SYSTEMS EFFECTIVENESS - NAVY

for presentation at

Summer Program for Industry
1964 Reliability and Industrial
Statistics Course

University of California at Los Angeles
10 August 1964

by

Keith N. Sargent
Head, Systems Effectiveness Branch
Office of Naval Material

United States
Naval Postgraduate School

THESIS

A PROPOSED STUDENT OFFICER INFORMATION RETRIEVAL SYSTEM FOR THE NAVAL POSTGRADUATE SCHOOL

by

Robert Lee Henry

and

Lawrence Lee Massa

June 1969

A Proposed Student Officer Information
Retrieval System for the Naval Postgraduate School

by

Robert Lee Henry
Lieutenant Commander, United States Navy
B.S., San Francisco State College, 1959

and

Lawrence Lee Massa
Lieutenant Commander, United States Navy
B.S., Naval Postgraduate School, 1964

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June 1969

Signature of Authors Lawrence L. Massa

Robert L. Henry

Approved by Richard K. Hamming
Thesis Advisor

Brooks J. Lockhart
Chairman, Committee for Computer Science

R. F. Rinehart
Academic Dean

ABSTRACT

Pertinent information on past and present Naval Postgraduate School students is now maintained, stored and processed in bulk files by Curriculum Officers. Information desired for management studies or analysis requires manual sorting of an ever increasing number of individual student records. This is an inadequate and inefficient system.

The foregoing problem could be resolved by the implementation of the Student Officer Information Retrieval System (SOIRS), which is a narrow scope retrieval system specifically designed to be responsive to the Curriculum Officer's needs with respect to student information. SOIRS evolved through a series of logical system design steps, identified as follows: (1) Problem Analysis; (2) Design of Records, Files and Reports; (3) Software Design; (4) Test of Entire System.

SOIRS is a complete system, establishing required files, updating files, and retrieving stored information.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF DEFINITIONS

1. Data Element - A cohesive unit of information within a record (e.g., the data representing the sex of an individual in a logical record would be considered a data element).

2. Record (Logical Record) - A collection of related data elements treated as a unit. (E.g., in an inventory control application, one line of an invoice is a data element and the complete invoice is a record.)

3. File - A collection of related records treated as a unit.

4. Entity - A person, place, or thing.

5. Information Attribute - A characteristic of an entity such as a person's name, curriculum, service number, etc.

6. Master File - A recording of the information about one set of entities of concern to the System (e.g., Current Master File is a file composed of records for each entity (Officer Student) currently attending the NPGS). There is one record for each entity.

7. Data Base - The total collection of operational data (data elements) in a system.

8. Batch Processing - A method of operation where transaction records are collected and system resources scheduled in such a way that the transaction file is processed against Master Files (in the case of SOIRS the master files would be the Historical and Current Files).

9. Binary Search - A nonsequential search of a table or list which is divided into two parts. A probe into such a list will yield one bit of information (i.e., the probe was either directed to the correct part of the list or it was not). Binary searches may have other binary searches as subsets, thus forming a binary tree search structure. Normally, the tree structure is developed to a level where short sequential searches of attributes can be conducted.

10. Bit - A binary digit representing the smallest manageable unit within a machine.

11. Byte - A sequence of eight adjacent bits; the most elementary addressable unit and is used to store one character (A,2,%,etc.,). In the system/360 a byte is reflected by base 16 notation within the machine:

the number $1_{10} = 11110001_2 = F1_{10}$

the letter A $= 11000001_2 = C1_{16}$

the left four bits of a byte are called the Zone Bits.

12. Boolean Masking - A bit or byte area of storage established at decision points within a processing program to compare conditions and alternatives based on boolean arithmetic. Logical "AND" and "OR" operations are typically used in masking schemes.

13. File-Activity Ratio - Defined as the ratio of the number of records used in one processing run to the number of records scanned in one run. In SOIRS this ratio could vary from a small number to 100% on a retrieval run, and will be 100% for all update runs. A large file-activity ratio implies the use of a high speed magnetic tape device. Tape will accomplish the processing in about the same time as using a direct-access device, and the former is a cheaper storage medium.

14. File-Volatility - Refers to the number of record additions and deletions to a file. In a static file environment, direct-access devices can be used with excellent results, however as the file-volatility increases these devices become quite inefficient due to a chaining process required to tie the file together. At this point a tape oriented file will provide a higher degree of efficiency.

15. Hashing Function - An algorithm which maps a set of keys into a set of integers, where the integers point to the specific storage location of the key related data.

16. Information Retrieval System - A process developed to recover specific information from a data bank.

17. Key-Searches - Searches conducted to retrieve information in accordance with a specific data input parameter (or set of parameters) called a key (search-key).

18. Key Transformation - See Hashing Function

19. NIBBLE - The righthand four bits of a byte. A NIBBLE is used in a great number of logical operations when the zone bits are not required.

20. Random Processing - A method of processing a file where the records of that file are not necessarily ordered in the same manner as the transaction records. A key transformation is utilized to direct the processing to the desired storage location.

21. Sequential Organization - A file organized in such a way that each record in the file is assumed to be placed in a series based upon an ascending sequence of some key field. In the case of SOIRS, this key field is the last name of students. The method of processing such

a file is termed Sequential Batch Processing (Sequential Processing), and is characterized by the condition that insertions and deletions of records to a file require rewriting the entire file. The transaction file is ordered in the same manner as the master file. Processing of the master file results in a serial transfer of records from the old master file to a new master file; inserting records on the new master and omitting the transfer of deleted records to the new master as directed by the transaction file. The obvious justification for such a file is a high file-activity ratio and a moderate (or higher) number of record additions and deletions.

22. Serial Search - An item by item key-search of a list (or file) until the search condition is satisfied, or the end of the list (file) is encountered.

I. INTRODUCTION

The maintenance and storage of pertinent information with regard to Officer Students presently attending the Naval Postgraduate School, Monterey, as well as those who previously attended, is currently the responsibility of the curriculum officer for each respective curricula. The bulk of this information is resident in one form or another in a manual storage media such as file folders, file cabinets or desk drawers. Over the years, the number of records accumulated is, in some cases, quite large, and consequently any retrieval of information required for one time reports, analysis projects, or recurring reports is a laborious undertaking. Additionally, the manual processing required to retrieve information might inhibit the initiative of an inspired individual who would otherwise attempt to use this data base for worthwhile analysis studies.

The authors have undertaken the project of developing a flexible and accessable automated student information retrieval system for the specific purpose of aiding the proper management authority in the execution of certain planning and control functions. The proposed system would be magnetic tape oriented and designed for implementation on the Naval Postgraduate School IBM/360-67 Computer. It would additionally be upward compatible with the IBM/360 family of computers from the Model 30 on.

The system as conceived could be utilized as a stand alone application which at a later date could be easily expanded to encompass the larger data base of a total personnel accounting sub-system, or it

could be integrated, with equal ease, into a total management information system. The goal of the design is to develop an information retrieval system which will assist the Curriculum Officer (or other designated managers) to carry out certain assigned responsibilities by providing needed information stored in a data bank, which could be retrieved easily when required.

The system to accomplish this is defined as one which would collect the data, update appropriate files in the data bank, and have the capability of retrieving any or all information from the data base.

In the conceptual stage of planning, the major tasks involved in the successful development of the system were identified:

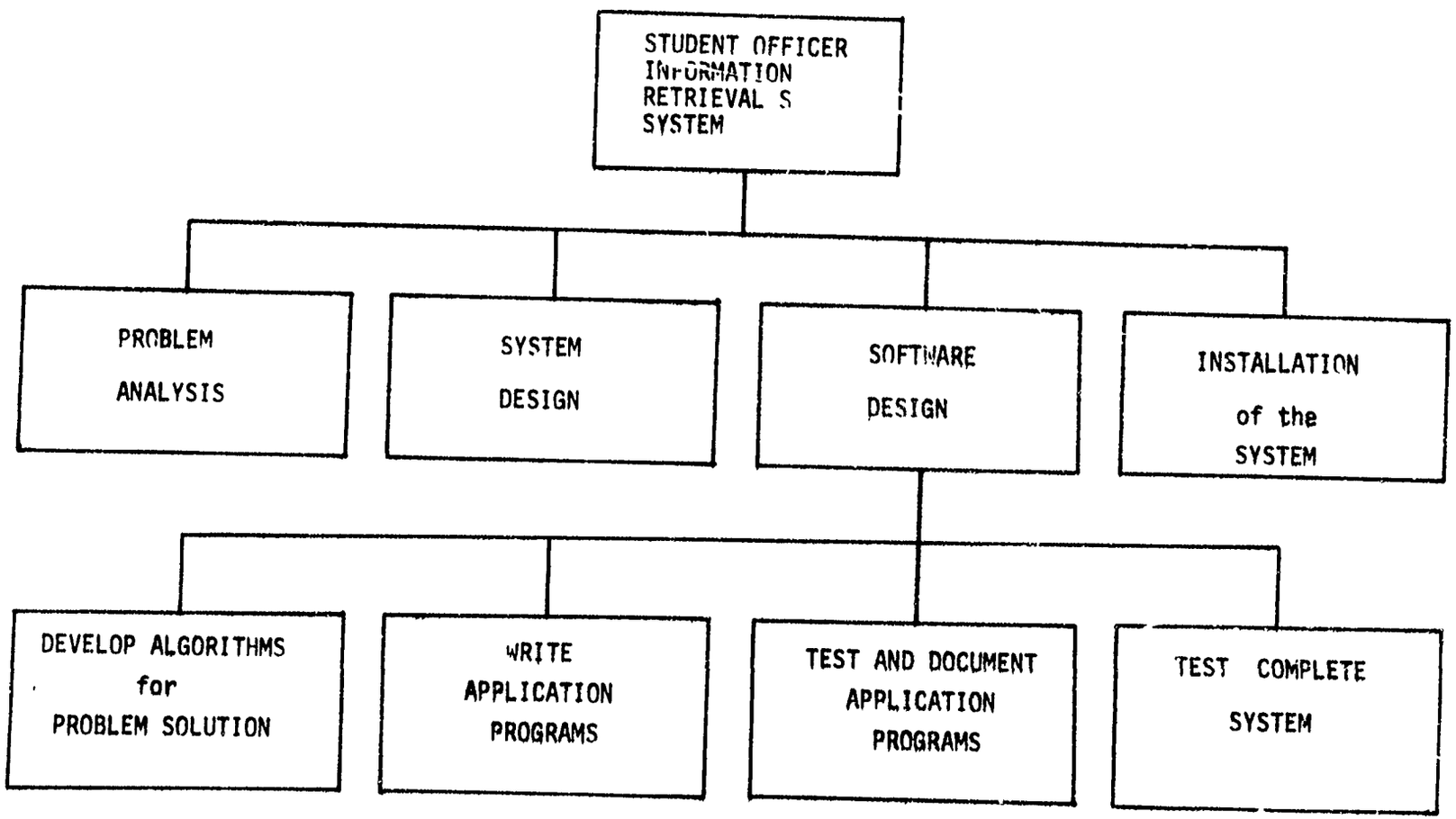
1. Problem Analysis (define the problem and the problem solution from the Curriculum Officers point of view).
2. System Design (develop the logic of the problem solution (algorithms), design the data base, and determine most appropriate method of file storage).
3. Software Design (write application programs for required algorithms, determine system programs to be utilized, and determine the programming language to be used for application programs).
4. Installation of the System (initialize the data base, develop user procedures, and document file status).

The foregoing tasks are illustrated in figure 1 together with a level of subtasks for Software design.

The remainder of this paper describes the evolution of the system development in steps corresponding to aforementioned tasks.

FIGURE 1

TASKS AND SUBTASKS
IDENTIFIED IN THE
INITIAL PLANNING PHASE



II. PROBLEM ANALYSIS

In the design of any information system, safeguards must be developed to insure that the system will be an effective management tool and responsive to the needs of as many managers as possible without overburdening the data base with information of marginal importance. The goal is to provide only that information which is necessary or highly desirable for management planning, operating, and control. Interviews with curricular officers were conducted to:

1. Insure the integrity of the proposed system;
2. Determine the most effective operational data for inclusion in the data base;
3. Ascertain the type of report information desired from the system.

Initially all Curriculum Officers were contacted and received a brief description of the proposed Information Retrieval System. At that time a list of tentative data to be included in the data base was submitted to the curriculum officers, and interviews were arranged for future dates.

The interviews generated the following list of necessary (or highly desirable) data which would be included in the data base:

1. File number
2. Name
3. Grade
4. Designator
5. Year group
6. Source code
7. Date of rank
8. P-code
9. Date of birth

10. Sex
11. Social security number
12. Expected retirement year
13. Curriculum number
14. Level of education
15. Designator change history
16. Date of arrival at NPGS
17. Graduate Quality Point Rating (QPR)
18. Total QPR
19. Date of graduation from NPGS
20. Degree area
21. Country
22. Service
23. Past duty station history

Interviews additionally indicated that it would be desirable to develop retrieval methods which would be responsive to information inquiries keyed to the following elements (or combinations thereof) of a logical record within the data base:

1. Designator
2. Curriculum
3. Grade
4. Date of birth
5. Sex
6. Level of education
7. Date of arrival (departure) to (from) the NPGS
8. Degree area
9. Country
10. Service
11. Graduate QPR
12. Total QPR

Based upon the Curriculum Officers expected use of an information system, it was determined that the foregoing requirements developed

during the problem analysis phase were realistic, and in most cases justified. Consequently, the design of the Student Officer Information Retrieval System (SOIRS) was initiated.

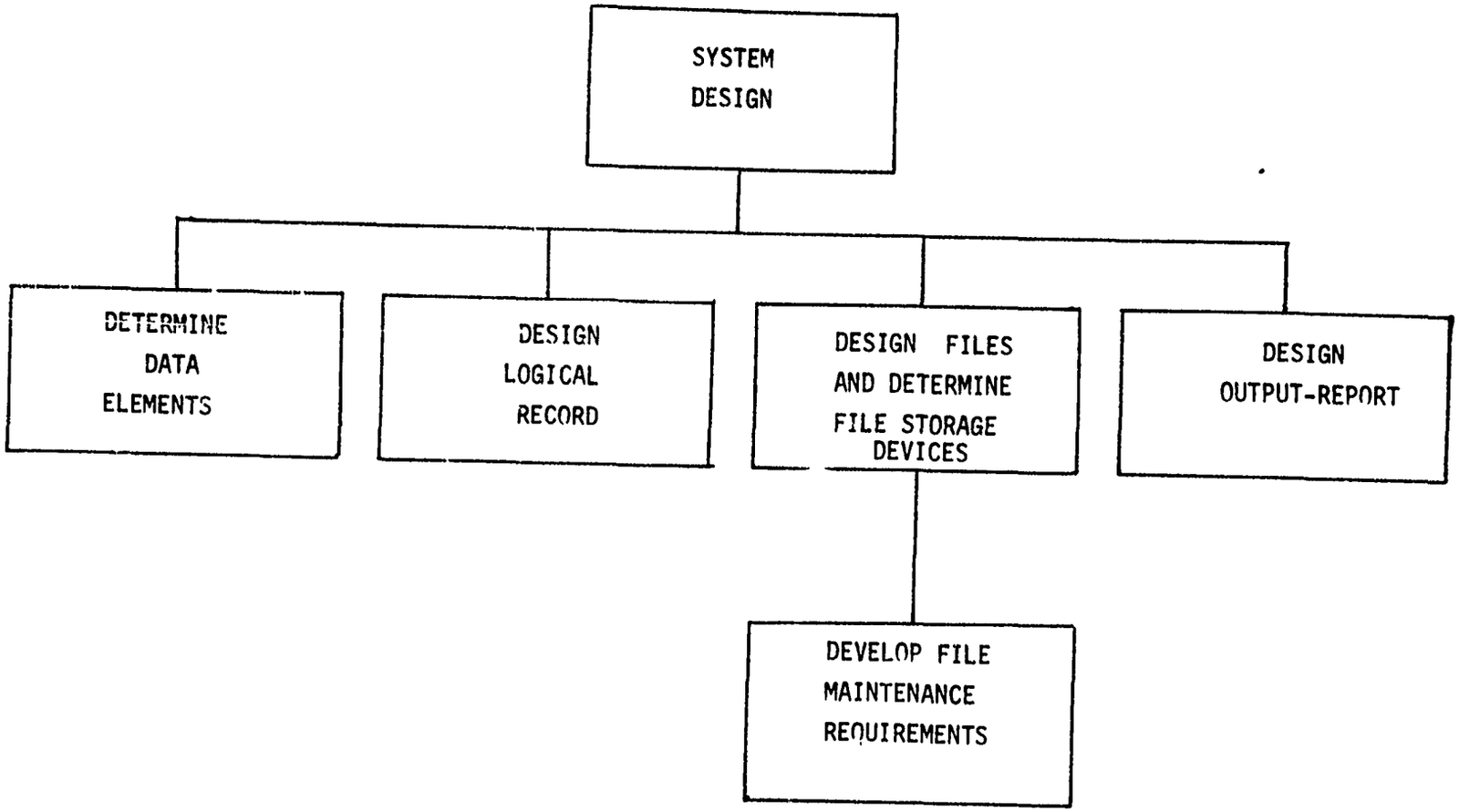
III. SYSTEM DESIGN

System design within the context of this paper will refer primarily to the development of logical record structure, file structure, file maintenance requirements, and report design. (Figure 2 presents the subtasks involved in this system design effort.)

The information content of an information processing system is given by its data base. Because of the extreme importance of information content the data base together with data files are planned, then the system is built around them (it is recommended that the reader refer to the LIST OF DEFINITIONS and read the entries "Data Element" through "Data Base" in the order presented). The determination of operational data for the data base was a two step procedure: (1) the first being locally generated at the NPGS as described in the section "Problem Analysis" and; (2) the second being the type of data which could be provided by the Bureau of Naval Personnel (BUPERS) to be used primarily in the development of a historical file of past NPGS students. BUPERS data relates only to officers in the U.S. Navy, whereas the NPGS generated additional data requirements due to the attendance of foreign students, allied officer students (other U.S. Services), and civilians. Unfortunately, an excessive amount of time was required in obtaining a BUPERS file, and the intersection of locally generated data elements and BUPERS data elements was not as large as desired. However, with the exception of two information attributes (undergraduate education; secondary P-code) all BUPERS data was utilized. The resulting information attributes for an entity within this system are depicted

FIGURE 2

SYSTEM DESIGN SUBTAS.'



in Figure 3, p. 20 (BUPERS did not provide information concerning date of arrival at NPGS, quality point rating (graduate or total), country, service, or history of past duty stations). All of the information attributes with the exception of Past Duty Station History are obviously useful within the concept of an information processing system. The inclusion of the current and seven previous duty stations was encouraged by the majority of people interviewed and justified by the desire to obtain information on the utilization of officer graduate education in duty station assignments, particularly in the technical areas. The record was designed before the realization that BUPERS did not provide the desired information; however, if the system is implemented, subsequent Bureau updates could be used to locally generate a duty station history.

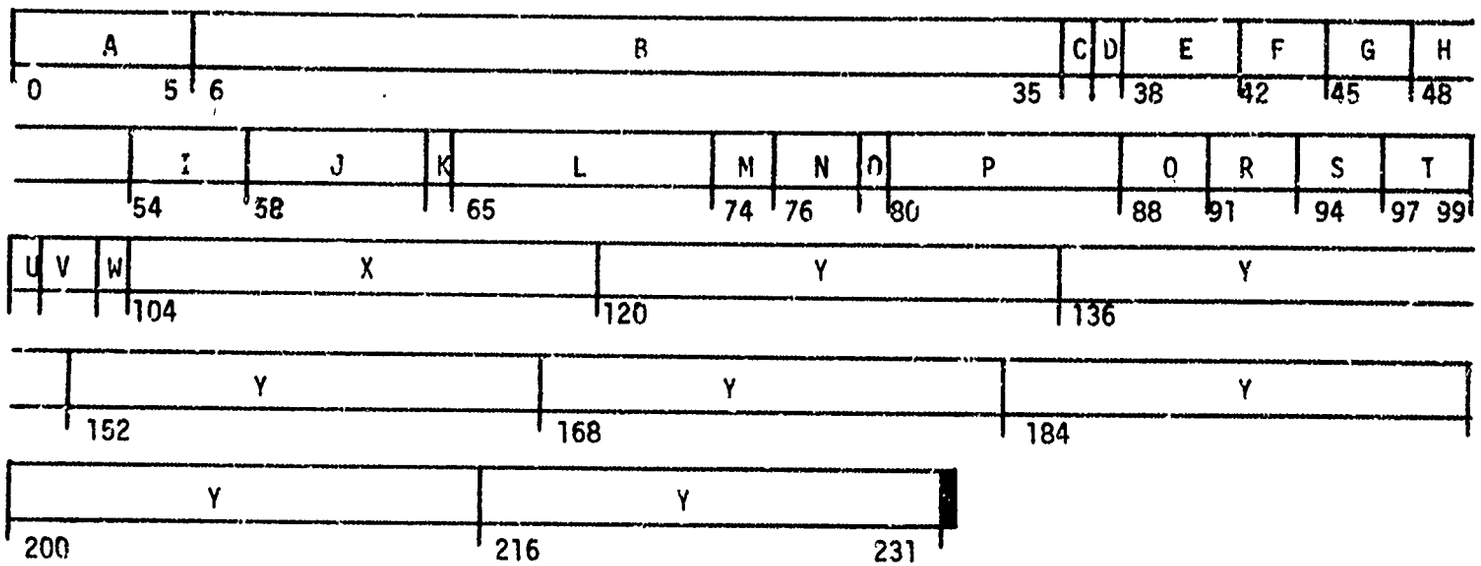
A. LOGICAL RECORD DESIGN

After determining the information attributes per entity, there was no difficulty encountered in assigning data elements to a logical record. The only specific requirements being: (1) insure that all data elements which might be utilized for key searches be contiguous and located at the beginning of the record in order to maintain a minimum length test mask, and minimize both the length of the data string and number of string manipulations required in search routines; and (2) provide space for a one byte test character expected to be utilized in update routines.

The logical record is presented in figure 3. Since all input to the system will be 80 byte card images (after initialization of the Historical File), the record naturally takes a form resembling 3 card

FIGURE 3

SOIRS LOGICAL RECORD



A. SERVICE NUMBER
B. NAME
C. SPECIAL KEY
D. RANK
E. DESIGNATOR
F. YEAR GROUP
G. SOURCE CODE
H. DATE OF RANK

I. P-CODE
J. DATE OF BIRTH
K. SEX
L. SOCIAL SECURITY NUMBER
M. RETIREMENT YEAR
N. CURRICULUM NUMBER
O. EDUCATION LEVEL
P. DESIGNATOR CHANGE HISTORY

Q. DATE OF ARRIVAL
R. DATE OF GRADUATION
S. OPR(GRADUATE)
T. OPR(TOTAL
U. DEGREE AREA
V. COUNTRY/SERVICE
W. NUMBER OF DUTY STATIONS LISTED IN X,Y
X. CURRENT DUTY STATION
Y. PAST DUTY STATIONS-listed in reverse chronological order

images with the last 8 bytes of the second image discarded. This latter condition exists to preclude the splitting of a data element on any card input record and is expected to reduce input errors. Data within all fields is left justified and trailing blanks are allowed in the name and station name fields. Some fields within the record may be blank, and these are described in detail in Appendix A, "Users Manual."

B. REPORT DESIGN FOR RETRIEVAL OUTPUT

As with the concept of an Information Retrieval System as a whole, the output report format was designed with the user in mind, adhering to the following guide lines as closely as possible:

1. A report should consolidate all information relevant to its intended use.
2. A report should be readable, uncluttered and attractive.
3. A report should be easy to interpret by the user.
4. A report should be helpful to the user in locating needed entries rapidly, with a minimum of confusion and searching.

Since all of the information attributes contained in a logical record were determined to be necessary, a report was designed to display all 232 bytes of information on standard printer output which permits the printing of only 132 characters per line. (A sample output format is illustrated in figure 7. Appendix A.) With the limitation imposed on the horizontal structuring of report information, a decision was made to use a column oriented output structure for the information attributes, stacking related attributes for each entity in columns when feasible. For example in figure 7., refer to the third column from the right hand edge of the report which is a stacking of two information attributes: DEGR (degree level, e.g., PhD) and AREA (the area in which

the degree was obtained). Note that these attributes are separated by a slash (/) which for the purpose of this report implies stacked output. Two other forms of stacked attributes occur in columns one and four. Column one information attributes are formated to correspond exactly to the column label appearing at the top of every page (those items in parenthesis appear only for selected entities for obvious reasons). Column four demonstrates the third and final method in which stacking is indicated: DESIG (current designator) appearing over (CHG), the latter refering to designator change history if applicable to an entity. The data printed for a designator change will appear in two stacked parenthesized expressions below the current designator; the first a four digit field indicating the year and month of the change, and the second indicates the old designator. Utilizing the foregoing methods of stacking, thirty information attributes can be efficiently contained in the fourteen column spread of the report output. The horizontal organization of the report as illustrated in figure 7 is basically column oriented with respect to individual or stacked information attributes. Even though abbreviations and pneumonics are freely used, each of the column headings are self explanatory and will pose no particular difficulty for report users (Curriculum Officers). In all cases, the headings appear directly over the data fields to which they refer and are printed at the top of each page of the report. Proceeding down the output report it is noted that double spaces separate the information printed (excluding any error notification) for individual entities in order to provide good visual effect and readability. Additionally, measures have been taken to ensure that page ejection will not occur once the printing of information for an entity has been

initiated, precluding the separation of related matter on adjacent pages. An additional feature of this report is the descriptive output for the information attributes RANK, DEGR, and AREA. In contrast with this output, refer to the Bureau output in figure 4 which contains these same information attributes buried as a subset in the second data field which is one continuous string of alphabetical and numeric characters. In this latter report the user would have to resort to tables or lists in order to decode the printed information. This is not the case with the SOIRS report. In certain cases, for U.S. Naval Officers only, reference to the Register of Officers might be required for designator or source code decoding; curriculum numbers are readily identified in the Naval Postgraduate School Catalog. An index to P-Codes is contained in current BUPERSINST 1210.13 series if this information is desired.

In the event an error is detected in the retrieval input data deck, the program will respond with an error message indicating that an error occurred in the data and that the data deck should be corrected and resubmitted. This error print-out should not appear on the report which is submitted to the report requester, but should be corrected by the individual conducting the file transaction. The events which could generate this error message are discussed in the sections "File Maintenance" and "Users Manual."

C. DESIGN OF FILES

There are two classes of entities included in the data base, namely those students currently attending the Naval Postgraduate School, and those who have attended in the past. This condition implies that a logical arrangement of two distinct files within the data base would

SAMPLE BUPLRS OUTPUT

KRUEGER RUDOLPH VINCE	667816457643282J1100630040		S PG MONTEREY	401019		U TEXAS	. 63 64300
KULESZ JAMES JOHN	647709219386251J1100610010		S PG MONTEREY	390827		USNA	- 61 64100
KUYKENDALL HERBERT BRENT	604124428528411I1100560011		WAR COL NEWPORT	310412		GWU	68 86700
LAACK DENNIS RAYMOND	648670394380147J1310410040		S PG MONTEREY	391225	63C61100	MARQUETTE	61 65600
LABYAK PETER STEPHEN	659217038265950J1100620010		S PG MONTEREY	390919		USNA	62 64100
LAIR RONALD JOHN	676454502446685J1320640380		S PG MONTEREY	410804	6700132565C61355	CONCDIA MH-	63 64400
LAKE ROGER ALLAN	719227535466512L3100680040		S PG MONTEREY	460621		U KANSAS	68 67700
LAMMERS LENNIS LARRY	641655516341930J1100600010		S PG MONTEREY	370716		USNA	, 60 64100
LANDRY ROBERT	688451267643708K1100690291		S PG MONTEREY	420105		U LOUISVIL	65E64400
LANGILLE JUSTIN EDWARD III	447614251644619G1100450010	9902P8803PHO	NAVMATCOMD DC	230204		RENSSELAER1259	N85400
LAPICOLA JOHN JAY	731732323388613L1100680010		S PG MONTEREY	460711		USNA	. 68 64100
LARSON DONALD RAY	563814577421121I1107570041	99000	S PG MONTEREY	320821	65C61105	AMERICAN U	62 76700
LASH FRANKLIN BRUCE	652595385347472J1100620060		S PG MONTEREY	381001	64C61105	U MINNESOT	61 68500
LASSITER REX DON	685002429705419J1320650382		S PG MONTEREY	390224	67101125670013 55	ARKANS SU	64 64300
LASSWELL JAMES BRYAN	669240567523988J1100630010		S PG MONTEREY	410925		USNA	. 63 64100
LATTA WILLIAM ALLEN JR	678325450640222J1100640010		S PG MONTEREY	420113		USNA	- 64 64100
LAUTERMILCH PAUL ANTHONY JR	543031177201213H1100510011	9801P9902PDCG	20 GMOROUGH	271111		GWU	- 64 86700
LAWRENCE GREGORY EDSON JR	508017570268965H1310500311	84010	INT STF JCS WASH	280815		AMERICAN U.	69 76939
LAWSON URBIE ALLEN	656512404495458J1100620040		S PG MONTEREY	390330		U LOUISVIL.	62 64400
LEAHY JOHN FRANCIS III	641859536300023J1100600010	83000	S PG MONTEREY	360525		USNA	. 60 64100
LEARDI PAUL LOUIS	649529223485621J1100610040		S PG MONTEREY	380804		U SO CAROL	61 68100
LEARY RICHARD KENT	733891004440497L1105680060		S PG MONTEREY	450927		BARRINGTON.	67 64300
LEBARON GEORGE ROHMER JR	699428568529635K1405460040		S PG MONTEREY	420718		FRESNO S	65 67200
LECROY FLOYD GARLAND	661509527308380J1400630291		S PG MONTEREY	300827		U MISSISSI	62E64300
LEE RONALD ALVIN	670319235507437J1400630060		S PG MONTEREY	400427	68031405	DUKE	. 62 64300
LENNOX RICHARD JOHN	669247139284970J1100630011	57035	S PG MONTEREY	380413		USNA	. 63 64100
LESEMANN DONALD FREDERICK	671084558507513J1400630060		S PG MONTEREY	390704	69031405	CAL POLY	62 60000
LETTINGTON THOMAS FLOYD	637637478383476J1100600040	96005	S PG MONTEREY	371216		IOWA STATE-	60 67400
LEVIN HELEN RUCKEL	550802355222546H3100550090		BUPERS PERS N	280504W		AMERICAN U	65 86900
LEWIS JOHN HOWARD	636552094302937I1100600060		S PG MONTEREY	360816	65061105	MANHATTAN	59 64300
LIEBERMAN JAY EDUARD	732845173368071L3100690040		S PG MONTEREY	460408		U MICHIGAN	68 65400
LIGON SAMUEL JOSEPH JR	630696579528500I3100590010		S PG MONTEREY	370814		USNA	- 59 64100
LILLY DAVID EDMUND	668659510327635J1100540271		S PG MONTEREY	310531	67001350	U WASHINGT.	63F65600
LINDER ISHAM WISEMAN	495352224527602G1310450010	9401P8103PLPD	7 CLEVEL	240214		U CALIFORN2461	R94301

FIGURE 4

provide a reasonable file base for the system. The file was designed in this manner and the two classes of files are refered to as the Historical File, and the Current File. There are several advantages in maintaining these files as separate entities; the more obvious are listed as follows:

1. After initial creation of the Current File, and subsequent to quarterly updates, each Curriculum Officer should be presented with a current listing of all students under his cognizance. This would preclude, except in rare instances, any further requirement for information retrieval from the Current File during that quarter.

2. The retrieval of information will normally be mutually exclusive with respect to the Current and Historical Files (i.e., keyed to either the Current File or the Historical File but not both simultaneously). For this reason the two file structure would be more efficient because fewer records would have to be scanned in each case. For example, the Historical File presently contains 4876 records, and the Current File will contain approximately 1500 records. The advantage obtained in searching only 5000 or 1500 records as opposed to 6500 records is obvious, especially when searching for current entities.

1. File Organization

Due to the absence of any real data regarding utilization of the information retrieval process, the criteria upon which the file organization is based are:

1. The expected number of record additions to and deletions from the files;
2. The size of files,
3. The number of records processed during update runs on the file.

The quarterly update of the Current File will involve a minimum of one information attribute (OPR) per entity to be updated, yielding a File-Activity Ratio (number of records processed divided by the number of records scanned) of 100%.

Addition and deletion of records with respect to the Current File are to be batch processed with the quarterly update, and are expected to number between two-hundred and four-hundred records

(the total number of arriving and departing students). Based on this, the Current File could in no way be considered static, but instead would be moderately volatile, justifying the rewriting of the Current File each quarter.

It is important to point out at this time, that there is presently no interface with Bureau of Naval Personnel Files which would permit updates of information attributes in the Historical File; this is a temporary condition pending implementation of SOIRS and is further discussed in Section V. Therefore, the term update is used with reference to the Current File only. However, records are updated and added to the Historical File every quarter when the Current File is purged of departing students.

The foregoing characteristics of expected Current File activity (high File-Activity Ratio and moderate file volatility), and the sequential nature of information retrieved from both the Current and Historical files indicate that a sequential file organization (refer to LIST OF DEFINITIONS) would be the most efficient for SOIRS, and was therefore utilized.

Normally, it would be inappropriate to initiate the information retrieval process with the specific intent of searching for one particular entity. On the other hand it would be quite appropriate to key a search to one or several parameters which would yield a list of entities within the intersection of search parameters. The latter use is the one for which the file was designed (in response to user desires), and with highly flexible retrieval techniques, the probability of obtaining information on any particular entity is uniformly distributed over the

length of the entire file. With the extensive range of search parameters provided it is necessary to scan each record of the file. Consequently it is a process quite compatible with sequential file organization. The alternate file organization would be one designed for random processing (see LIST OF DEFINITIONS) with searches established through the use of hashing functions. This would be extremely inefficient and time consuming due to the complexity of key transformations and an increased number of logical operations necessary to acquire the desired degree of flexibility.

2. File Residence

The decision resulting in a sequential file organization yielded two alternative devices upon which the data files could be stored and manipulated: (1) IBM-2311 disk pack or, (2) magnetic tape. The criteria considered in choosing magnetic tape as the appropriate device are listed as follows:

1. Cost
2. File protection
3. File security
4. File compatibility
5. Device speed

Cost in this context refers to the cost of data residence on a particular storage device. The cost for magnetic tape is approximately .01¢ per bit and the disk cost is approximately .1¢ per bit.[1] This latter figure is deceiving since, for best operation,

---

[1] Professor M.L. Cotton during a course of instruction concerning Information Structures, March 1968.

the entire disk pack should be dedicated to the subject files, and for the cost of one disk pack, several magnetic tapes could be purchased (these tapes would exceed the disk storage capacity many times over).

File protection refers to the vulnerability of the storage media to inadvertant destruction during file maintenance routines or testing. Both tape and disk provide for the unique labeling of a file to discourage unauthorized use, however, of greater concern is the desirability to maintain processing control over the stored data.

This latter control is best illustrated by an example: Suppose that during a file maintenance or testing routine, which required a file to be read, the completion of the data control block was in error due to incorrect use of OPEN MACROS, DCB MACROS or Data Definition statements which would result in a write operation. This could generate processing which would destroy a file unintentionally. In the case of magnetic tape, positive control over a read/write operation can be exercised by removing (or replacing) the "WRITE RING" from (in) the tape reel. When the ring is out the tape cannot be written on; only by inserting the ring can a write operation occur. This should significantly decrease the potential vulnerability of a file and preclude a lengthy process of file re-creation. The disk storage media does not provide this feature of protection.

File security refers to the physical security of a storage media which might or might not contain sensitive information. Magnetic tape is highly portable, moderately durable, and small enough so that several reels could be stored with ease in a safe or small vault. If required, tape reels could be transported to a computer

center, remain under observation during a processing run, and upon completion of processing returned to the designated storage area. The disk pack although portable, is fragile and bulky by comparison; additionally, disk packs are not normally removed from the processing center.

File compatibility refers to a decision to use either magnetic tape or disk storage or both for different files. The Grandfather, Father, Son concept of file back-up is utilized for both Historical and Current files to permit recreation of any file which might be destroyed in error. The most practical method of storage for these generations of files is magnetic tape for reasons mentioned in preceding paragraphs. In order to promote compatibility in file devices and to minimize the data management problem for the programmers who will maintain the file, magnetic tape was used for all files. (Additionally, magnetic tape is a universal storage media which is compatible with most computer installations; this is not currently the case with disk files.)

Device speed in this case relates only to the transfer rates of the respective devices; IBM-2311 has a transfer rate of 156K bytes/second and magnetic tape devices have a transfer rate of 90K bytes/second. Clearly a distinct advantage in favor of disks; however, with the advent of multiprocessing with a variable number of tasks (MVT) presently available on the Naval Postgraduate School computer. This transfer rate was not considered significant since the additional input/output requirements expected with tape would not adversely affect the utilization of the central processing unit.

D. FILE MAINTENANCE

For the purpose of this discussion, File Maintenance will include a brief description of Historical File creation, file transactions involving input data which result in changes to files, and a proposed method for insuring the integrity of files.

1. Creation of Historical File

The following methodology describing the generation of a Historical File is applicable to the development of files which are based (whole or part) on data received from an outside activity, and would be appropriate in subsequent dealings with the Bureau of Naval Personnel in the event that SOIRS is implemented.

Extreme caution must be exercised in dealing with a tape received from outside activities since incorrect processing could result in the loss of tape data, requiring additional transactions, time in shipment, and computer time at both the sending and receiving activities. A request to BUPERS resulted in the receipt of a magnetic tape file which contained records on all past and present Navy students of the NPGS. The tape as requested was a labeled tape with a track density of 800 b.p.i. Upon receipt the tape label was verified by utilizing the IBM System Utility "IEBPTPCH" to read and print the label. Subsequent to label verification the entire content of the tape was printed out, employing the same utility, providing visual assurance that the transmitted file contained desired information. Immediately thereafter a working copy of the BUREAU tape was generated, with appropriated blocking factors for economical buffer utilization,

through the use of the System Utility "IEBGENER." It was previously decided that SOIRS files should be arranged in alphabetical order (first 5 letters of the last name) for batch processing, and to additionally facilitate easy reading of the output report. In order to achieve this ordering, the System Procedure "SORT/MERGE" was utilized. The results of the foregoing processing yielded a sorted working copy of the Bureau file ready for processing with the SOIRS application program "BUPERS TRANSFER" (BUPSXFR). The BUPSXFR program essentially takes a record from this working copy, abstracts desired information attributes, performs any data conversions required, and places these attributes in a SOIRS logical record format. As the record for each entity is transformed, it is written onto a tape which, when completed, will be the Historical File. This processing program does not convert or place on the Historical File any Bureau record which does not reflect past attendance at the Naval Postgraduate School. Instead, the program causes the first 133 bytes of these rejected Bureau records to be printed out. Upon completion of processing, the Historical File has been generated and a listing of rejected records is printed out (see figure 4 for sample rejects).

2. File Transactions

There are two other SOIRS application programs which will be regularly used in file transactions. These are the CRDCHK1 and UPDATE programs, and are discussed in the following paragraphs.

a. Input Record Verification

Upon initial creation of the Current File and for subsequent insertion of logical records for new entities, the CRDCHK1 program will be utilized to insure the correctness of card input records

prior to placing the subject records on the Current File. The following attributes of each input record are checked, as appropriate, for correct numeric and alphabetical strings, incorrect use of blanks or blank fields and special characters, and for correct ranges of both numeric and alphabetical fields:

1. Name
2. Number of duty stations
3. Designator
4. Special Character (*)-required in column 37
5. School Start Date
6. Curriculum
7. Degree
8. Graduate and Total QPR
9. Rank
10. Service Number
11. Sex
12. Expected Graduation Date
13. Area
14. Social Security Number
15. Service and/or Country

Appendix A contains a sample CRDCHK1 output with a list identifying the types of errors which will be reported.

b. File Updates and Transfers

It was determined that all file updates and transfer transactions could best be accomplished using batch processing techniques on a quarterly basis. These transactions are:

1. Updating records in the Current File
2. Inserting new record in the Current File
3. Deletion of records from the Current File
4. Transferring the updated records of those students who have completed their tours at the Postgraduate School from the Current File to the Historical File.

The foregoing transactions are incorporated into one processing program, UPDATE, which performs these functions in a single processing run conducting updates and record dispositions as directed by input data cards. The update parameters and special character keys required in the input data are described in detail in Appendix A.

The Current File should at all times be maintained in alphabetical order (first five letters of last name), and since batch processing is the designated mode of file transacting when using UPDATE, the input data cards (less the record additions) should also be sorted in a like manner. An out of sequence data card will not be processed, but will generate an error message which identifies the card in error. The program will recover and proceed to process the next record. Other types of errors in input data are handled in a like manner. A sample listing of UPDATE output, which illustrates the four basic types of program response, is contained in Appendix A (figure 11). and is described as follows (the entities in the following examples may be located on the output listing by proceeding alphabetically down the name field):

1. The first transaction involves the updating of the entity "ACADEMY, JOE FRESHCAUGHT." The first line is a portion of the logical record prior to any changes, and the second is the updated record as it will appear in the new file. The transaction is identified by the suffex * UPDATE* appearing after the updated entry.

2. The second entry, "ARMYTYPE, ALPHONSE KNOTHEAD" has a suffex of * DELETE *, indicating that the record for this student has been deleted and will not appear on the new Historical or Current File.

3. The output for "FLEGLE, OOGLE EYE" has a suffex of * XFER *, which implies two conditions: (1) the record was transferred from the Current to the Historical File; and (2) prior to the transfer any required updates to the subject record were completed. This output is similiar to that obtained with * UPDATE *, demonstrating the before and after conditions of the record.

4. The final transaction, "LUCKY, GO HAPPY", illustrates the last type of UPDATE output, and is identified by the trailing * NEW * in the one line output. This output constitutes the first 122 bytes of the new record. (Input data cards for new records are previously checked for accuracy by processing with the CRDCHK1 program).

The UPDATE printed output additionally becomes a useful medium for conducting spot checks on desired transactions.

3. Quarterly File Maintenance

The most important consideration in any information system structured on a dynamic data base is the protection of that data base from accidental loss or destruction. In the case of SOIRS, the two basic files, Historical and Current, should be maintained as described in following paragraphs in order to preclude the total loss of the data base, and to provide for reconstruction of both files to their most current form.

Subsequent to SOIRS implementation, the file content will consist of one copy each of the two basic files. Backup copies of each file should be generated as soon as possible by employing the System Utility "IEBGENER" (a back-up copy of the Historical file exists at the present time), yielding two copies of each file which are in turn sorted in alphabetical order. One each of the Historical and Current File copies will become working files, while the others are to be designated the "Current Father" and "Historical Father" files respectively. These fathers therefore are the backups for the working copies.

Quarterly, the UPDATE program is used to process record updates, deletions, additions, and transfers to the Historical File. All of these transactions are accomplished in a sequential manner which does not affect the working file, but instead performs the designated

operation and writes the new record on a new Current File. In the case of transfers, the record is written on a new Historical File. Upon completion of UPDATE processing, the new Current File contains new record additions, updated records for all entities which were resident on the Current Father and were not transferred to the Historical File, and does not contain records which were designated as DELETE from the Current Father. This new Current File is then sorted and becomes the Current Son, a new generation of current entity records. The UPDATE input data records should be retained for one quarter so that the Son could be regenerated from the Father, by another UPDATE processing, if required. The new Historical File is generated through UPDATE transfer of designated records from the Current Father to the new Historical File. After UPDATE processing, the Historical Father is copied onto the new Historical File (using IEBGENER with a disposition of MOD, KEEP). The new Historical File contains all records transferred from the Current Father and all records which reside on the Historical Father. Sorting of this new file yields the Historical Son; a new generation of historical entities. The creation of subsequent Current and Historical generations are carried out in the same manner as described above. The retention of files one generation old, and the retention of quarterly transition data for the UPDATE program provides for adequate protection and restoration of the data base in the event of destruction of the Current and/or Historical Sons.

IV. SOFTWARE DESIGN

The program documentation for each of the application programs is presented in Appendix B in the form of flow charts and program listings. Each line of instruction in the listings contains, in the comment field, a descriptive phrase concerning the logic of the operation, and constitutes a highly descriptive mode of documentation which will be extremely valuable to application programmer assigned responsibility for program maintenance. The following paragraphs will briefly describe in less detail certain characteristics of specific algorithms used in each of the problem programs.

A. BUPSXFR PROGRAM

This program was developed to:

1. Read the BUPERS logical records.
2. Test each entity for validity with respect to Historical File criteria.
3. Perform data conversion on certain attributes.
4. Transform required attributes to SOIRS logical record configuration.
5. Write appropriate records on the Historical File.
6. Print out a listing of those records which were not placed in the Historical File.

With the exception of 2,3, and 6, above, all programming techniques are elementary in execution and require no special knowledge on the part of the reader with regard to the BUREAU logical record. These exceptions (also employing only basic programming techniques) are listed and described as follows:

1. The criteria for entity placement into the Historical File is that a student must have attended the NPGS in the past. This condition is

ascertained by comparing the character constant "MONTEREY" with the college name attribute of the BUPERS record field "Education 1." An equal compare implies, that this record should be placed in the Historical File.

2. Upon receipt of the BUPERS File, it was found that information concerning course curriculum was not included in the file and information regarding degree area was encoded in accordance with the "Naval Officer Education, Language and Service Schools Code (NAVPERS 15824 series)." The latter being a rather extensive numeric listing of specialty codes which exceeded the requirements of SOIRS (These requirements are indicated as a list of twenty attributes in TABLED of the Information Retrieval Program, Appendix B.). In order to convert the Bureau data into meaningful SOIRS attributes, a table was constructed which mapped the four byte Bureau data concerning education onto curriculum numbers and degree area. This table is labeled as CURRTAB in the BUPSXFR listing, Appendix B. All of the 79 attributes of CURRTAB are organized in the same manner. The first four numbers of each CURRTAB attribute indicates a Bureau education code. The next three numbers indicate the curriculum onto which the Bureau code is mapped, and the letter indicates the degree area, which is the final attribute in the range of the mapping. As an education code is read from the Bureau tape a binary search of CURRTAB is conducted for a match on the first digit of the code. When an equal compare occurs, the pertinent subset of CURRTAB is serially searched until a match on the four digit code is found; then the data is converted yielding a curriculum number and degree area for each entity. This binary search is relatively effective, requiring at the maximum a serial search of a 26 item list (from a total list consisting of 79 items) to obtain the correct entry in the table. This maximum occurs only when the last item in the 5000 series of CURRTAB is required (see figure 5).

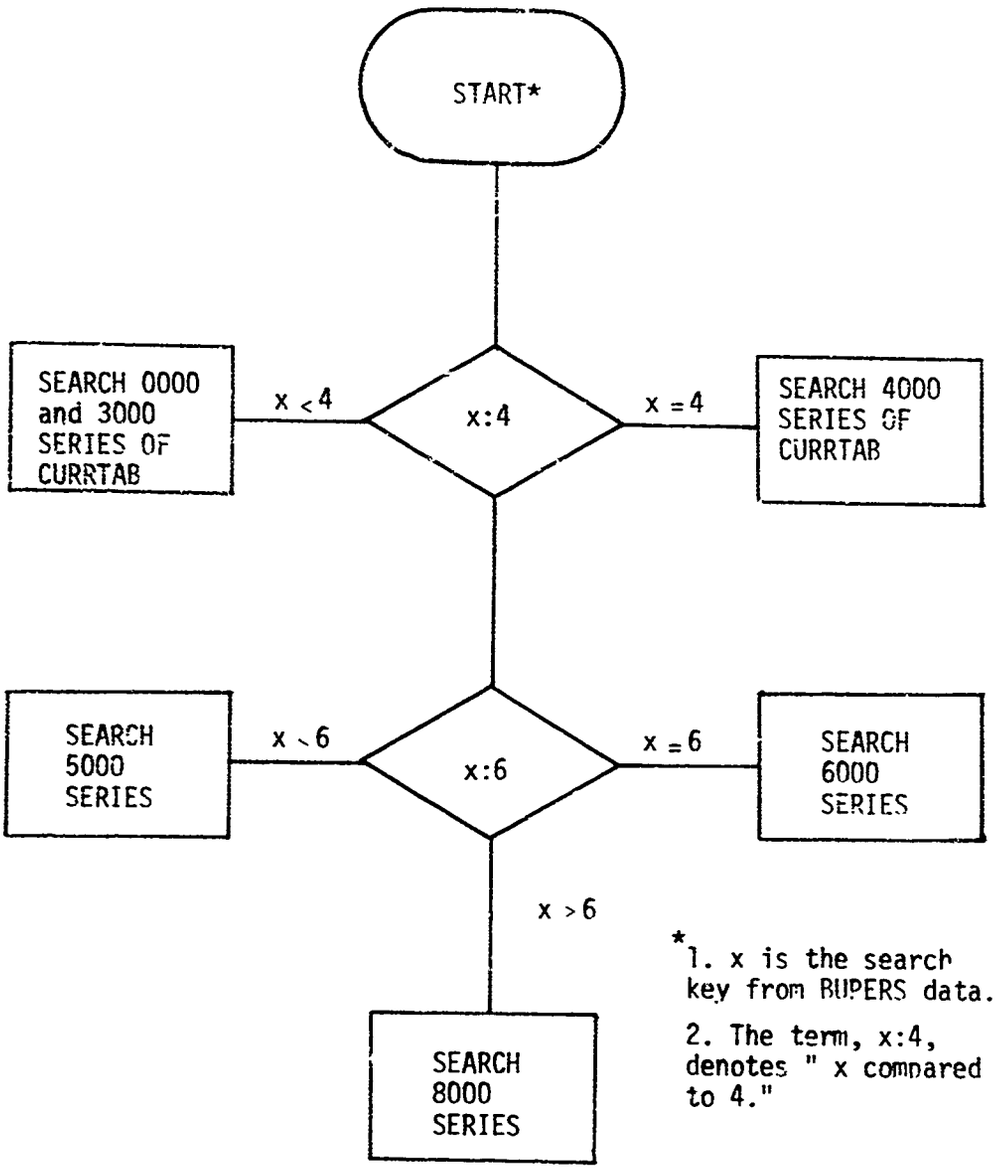
3. If the criteria in item one above is not satisfied, or if the binary search of item 2 does not yield a correct response the program will reject the record and cause its contents to be printed on the output printer (i.e., the records printed are not included in the Historical File).

B. CRDCHK1 PROGRAM

This program was designed to verify the correctness of SOIRS input data cards (for new entities) destined for processing with the UPDATE program (the general program objectives have been previously described in FILE MAINTENANCE section III (D)(2)). The program specifically checks the following information attributes (from the SOIRS logical record) for the errors indicated:

FIGURE 5

BUPSXFR BINARY SEARCH OF TABLE CURRTAB



1. Name - The first byte is checked for a blank, and an equal compare results in an error. The first ten bytes of the name field are checked for correct alphabetical characters and if numeric or special characters are detected, an error exit occurs.

2. Duty Station Number - This byte is checked for a numeric value in the interval (0,8), any other character will result in an error condition. Up to three duty stations can be indicated on the first two data cards; if an entity had 3 or less duty stations and an incorrect number (greater than 3) of duty stations indicated, the program would automatically assign a duty station number of 3 to correct the error. During processing, a zero (0) is inserted in this attribute for each foreign entity to serve as a potential key in future transactions.

3. Designator - Each byte of this four-byte field is checked for numerics on the interval (0,9), any other characters result in an error exit.

4. Byte 37 of the input record is tested for a *, which must be present in order to convey condition codes to the UPDATE program, otherwise an error results.

5. School start (end) date, a three byte field, is checked for numerics on the interval (0,9), any other character will result in an error.

6. Curriculum Number - a three byte field indicating the student's curriculum is checked for numerics on the interval (0,9). Other characters result in errors.

7. Educational Level, Graduate and Total OPR, Service Number, and Social Security number are checked for valid numerics, on the interval (0,9), for each byte of the respective attribute.

8. Rank, and Degree Area are one byte fields tested for the alphabetical characters on the interval (A,T). Any other character results in an error condition.

9. Country/Service - A two byte attribute which indicates the branch of service for U.S. students, and indicates the country of a foreign student. The first byte must be numeric on the range (0,4) and the second byte numeric on the range (0,9). Both bytes collectively have a maximum permissible value of 41. Deviations from the foregoing result in error exits.

10. The attribute SEX is checked for one of the following valid characters F,M, or a blank. The latter two indicating "male." All other characters result in errors.

Errors generated from the above processes are descriptively printed out identifying not only the attribute name, but the character which caused the error. A complete description of all errors is contained in Appendix A.

1. CRDCHK1 Algorithm for Table Output Processing

The CRDCHK1 program contains five tables (C,D,G,L, and S) which assign meaningful output parameters to the record attributes which exist as one or two byte table keys in the logical record. For example, TABLE L of the CRDCHK1 program lists the ten possible levels of education which are considered in SOIRS. Any one of these levels may be indicated in the logical record image by inserting a number ranging from 0 to 9 in byte 101 of the record. When this byte is processed by CRDCHK1, this numeral must be transformed to a meaningful mnemonic or abbreviation which will appear in the output report. The manner by which this transformation occurs is described in the following example, using TABLE L for illustration.

address (b_{10})	TABLE L	Table key
0000	GSCH	0
0004	HSCH	1
0008	1YRC	2
0012	2YRC	3
0016	3YRC	4
0020	-PG-	5
0024	BACH	6
0028	CERT	7
0032	-MS-	8
0036	PHD-	9

Hexidecimal and binary formats are used for representing all characters internal to the machine, but for purposes of this example will be supplemented by base 10 notation.

The algorithm takes the hexadecimal representation of the Table Key and logically "ANDS" the Key with the hex constant "0F." This results in a pure binary number which is then multiplied by four (4). Adding this result to the address of the first attribute in TABLE L yields the address of the desired attribute. For purposes of illustration

let the table key be equal to $2_{10}(F2_{16} = 11110010_2)$. Submitting this value to the logical "AND" operation with $0F_{16} = 00001111_2$:

$$\begin{array}{l} 11110010_2 \\ \underline{00001111_2} \\ \text{result of "AND"} \quad 00000010_2 \end{array}$$

multiplication by 4 yields: 00001000_2

adding this value to the address of the first attribute of TABLE L:

$$\begin{array}{l} 00000000_2 \\ \underline{00001000_2} \\ 00001000_2 = 08_{10} \end{array}$$

where the address 08_{10} is pointing to the desired attribute 1YRC (one year of college)

In the example above, the address of the first attribute of TABLE L was assigned a value of 00 only for convenience in computation. During actual program execution, this address can be located at any authorized absolute address in core storage. The advantage of this method of table retrieval, is that the desired attribute is obtained in one table reference as opposed to a time consuming serial search of the table, potentially involving all attributes. The attributes of the four other tables in CRDCHK1 are obtained in a similiar manner using the same basic algorithm.

C. UPDATE PROGRAM

The General Characteristics of the UPDATE Program are described in section III. D. 2.b; the basic functions of the program are reiterated as follows:

1. Add new records to the Current File

2. Update records in the Current File

3. Update and transfer records from the Current File to the Historical File.

4. Delete records.

The UPDATE Program permits updating of the following information attributes only:

1. Designator
2. Designator change history
3. Rank
4. Date of rank
5. Curriculum number
6. Degree area
7. Degree level
8. QPR (graduate and total)
9. Graduation date.

As each input data card is read, a test is performed to determine if the entity is a U.S. or Foreign Officer. A star (*) in column 1 indicates a Foreign Officer and the file is searched alphabetically until a match is encountered. If column 1 is blank a serial search on social security numbers is conducted until the desired U.S. Officer record is found. The following parameters on input data cards indicate record disposition:

1. A percent sign (%) indicates delete the subject record from the file.

2. A dollar sign ($) indicates the record is to be updated and transferred to the historical file.

3. A star (*) indicates a new record addition.

4. The absence of any of the aforementioned special characters results in an update to the subject record and retention in the Current File.

Additional requirements for input data preparation are discussed in detail in Appendix A.

D. FILESRCH PROGRAM

The Information Retrieval Program (FILESRCH) is at the hierarchical apex of all SOIRS programs and is the essence of the system. FILESRCH is a management tool and will respond to information inquiries keyed to the following information attributes subject to the search-key parameters (equal (=), not equal (#), and an inclusive range (%) of two numbers (2 digits each)).

Possible Search Attributes	Possible Search-Key Parameters
Social Security Number (A)	=
Sex (B)	=
P-Code (C)	=,#
Designator (D)	=,#
Service/Country (E)	=,#
Year of Birth (F)	=,#,%
Year Group (G)	=,#,%
Rank (H)	=
Curriculum (I)	=
Degree Area (J)	=
Degree Level (K)	=
OPR (Graduate) (L)	%
OPR (Total) (M)	%
Graduation Year (N)	=,%

(The alphabetical characters in parentheses after each attribute, A-N, are keys used to refer to respective attributes during input data preparation.)

The result of FILESRCH program execution will yield a subset of the file searched corresponding to the logical "AND" of each search attribute (together with its search-key parameter) with all other search attributes desired. The following two elementary examples will illustrate this property.

1. A file search keyed to the following inputs: B=F, F%36,45, I=360, L%25,30 will yield a list of Female students with a year of birth ranging from 1936 to 1945, who were enrolled in curriculum 360

and possessed a graduate QPR ranging from 2.50 to 3.00. The entities for which all of the foregoing attributes are true would appear on the SOIRS output report.

2. A file search keyed to: A=549426211, D=1100, F#40 would yield at maximum one entity; the student who has a social security number equal to 549-42-6211, a designator equal to 1100 and who was not born in the year 1940. (Obviously, since a social security number is unique, the output list could contain no more than one student, and then if and only if all search attributes are true). This case is an example of a search which should not be initiated since the information on one individual is easily obtained from a quarterly list which should be made available to each Curriculum Officer.

During a given retrieval run it is possible to key on all search parameters (A-N), however, as indicated in example 2 above, certain combinations could result in absurd output listings. The limitations on searches are: (1) each search attribute can be used only once for each program run; and (2) the range search-key (%) can be used only twice during a program run. This latter restriction arises from a programming requirement to use only four bits to set condition codes. Two bits are required for = and #, and each % requires one additior bit; in this case a maximum of two range checks are allowed in order to contain all testing criteria in NIBBLE configuration. Even with the above restriction, there are 6.24×10^5 possible independent searches available resulting in a rather high degree of search flexibility.

1. The Basic FILESRCH Algorithms

FILESRCH will process desired information in accordance with the type of Search-Key Parameters (=,#,%) specified in the input data. To keep track of seven possible combinations of search-keys, a one byte storage area called FLAGMASK (FM) is established to record and subsequently direct the FILESRCH processing with respect to search-key requirements. FLAGMASK will have one of the following configurations after all input data has been processed:

FLAGMASK	Search-Key Processing Required
111_2	%,#,=
110^2	%,#
101	%,=
100	%
011	#,=
010	#
001	=

The algorithm which directs correct search-key processing is illustrated in figure 6 (normal use of Immediate Compare and Test Under Mask instructions are required).

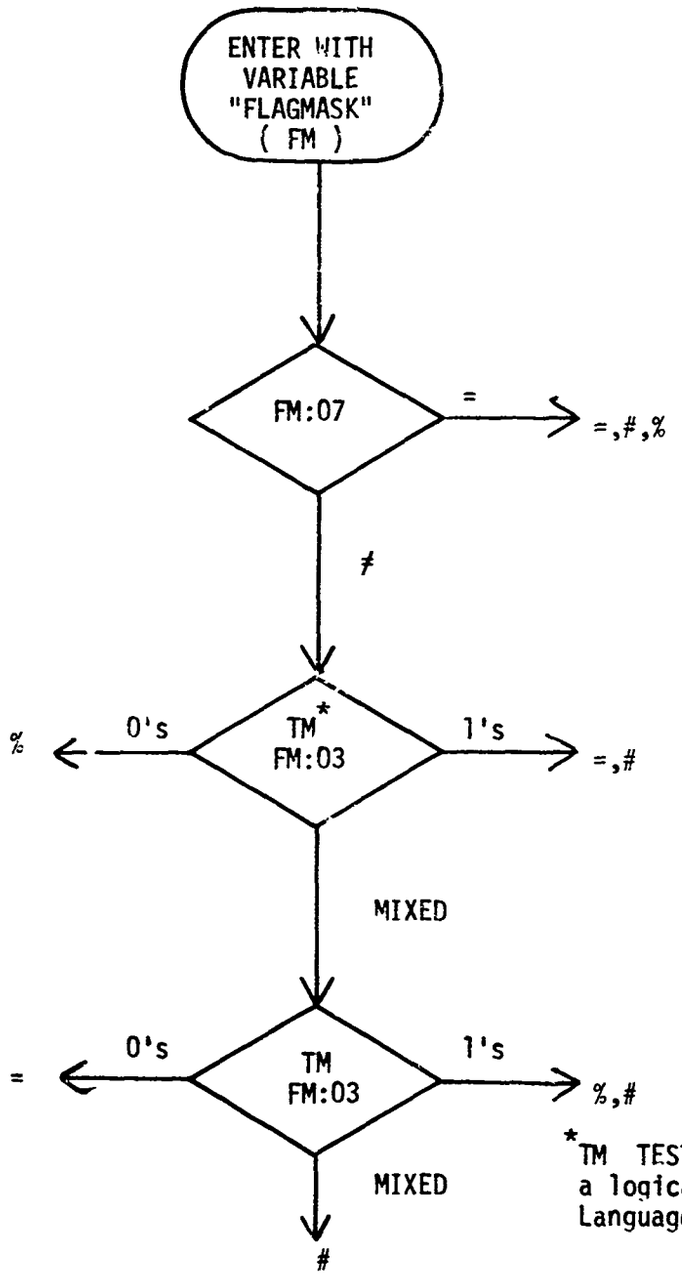
Depending on the FLAGMASK setting, input data is compared with each record of the file and appropriate entities are listed in the output. The method of introducing desired information attributes to the processing program will be discussed in the following paragraphs.

All of the possible search attributes are contained in bytes 37-103 of SOIRS Logical Record. This results in a 67 byte field which might contain one or more sub-fields of interest. In order to extract the required attributes, a masking scheme was developed for each key-search parameter (%,=,#). For purpose of discussion, only the equal key parameter will be referenced.

Three 67 byte storage areas are utilized to produce a field which will be used in a serial search of the file. For purposes of illustration these three areas are to be called EQMASK1, EQMASK2, and SAVEQ. When an input data card indicates a key-search parameter of equal (=), the search attribute is placed in EQMASK1 (initially all zeros) in a position corresponding to the attribute's location in bytes 37-103 of the logical record. One's are inserted into the corresponding bytes of EQMASK2 (initially all zeros). This process

FIGURE 6

DECISION ALGORITHM for DETERMINATION
of SEARCH-KEY PARAMETERS



continues until all input data cards are processed. At this time bytes 37-103 of the first file record are placed in SAVEQ. Immediately a logical "AND" between SAVEQ and EQMASK2 is executed resulting in destruction of undesired attributes in SAVEQ. EQMASK1 and SAVEQ are then compared character by character. If an equal compare results, the subject record is desired and printed. For example, let the following be indicated on a data card as the only desired information: B = F, where F = 11000110_{16}. The user desires a list of all female students. The hexadecimal equivalent of "F" would be placed in byte 28 of EQMASK1 and ones would be inserted in byte 28 of EQMASK2:

EQMASK1	0 ← zeros	11000110 byte 28	zeros →	66
EQMASK2	0 ← zeros	11111111 byte 28	zeros →	66

Reading the first record from the file and moving the 67 bytes of possible search fields to SAVEQ, would yield in the case of a male entity:

SAVEQ	0 ← other data	11010100 byte 23	other data →	66

"ANDING" SAVEQ with EQMASK2 yields:

SAVEQ	0 ← zeros	11010100 byte 28	zeros →	66

A compare of EQMASK1 and SAVEQ will result in the rejection of the current record. This process would be continued until the file was exhausted. The output report will contain all female students.

Based on the configuration of FLAGMASK, the other six methods of processing would be conducted in a similiar manner.

A complete description of FILESRCH input data preparation, together with several examples, is included in Appendix A.

There is one final characteristic of the FILESRCH program which will be valuable to users. This is the inherent ability of the program to dump the entire file without resorting to the System Utility IEBPTPCH. This may be accomplished by using one of several search attributes and specifying a search-key parameter of unequal (#) with data which could not be included in any record. For example, search the file for entities who were not born in the year 99, or those who have a social security number not equal to 999-99-9999. The results would obviously yield a listing of the entire file.

E. CHOOSING A PROGRAMMING LANGUAGE

The most important criteria leading to the choice of Assembly Language (AL) as the programming language for SOIRS, was the desire to achieve maximum compatibility with the Bureau of Naval Personnel which employs AL for Officer Files. The only alternative languages considered were those from the List Processing Set (LISP, PL-1, etc.,), and those were inferior to AL with respect to efficiency of machine execution. Additionally, AL is more commonly used (locally) than any list processing language, and it is therefore expected that program maintenance will be carried out more effectively.

V. CONCLUSION AND RECOMMENDATIONS

This paper has discussed an Information Retrieval System which was developed to fulfill certain specific requirements for a particular group of managers. As noted in section II, the desires of these managers resulted in designing a narrow scope retrieval system. However, adequate space has been provided in the logical record to expand the number of information attributes per entity if desired, and there is sufficient modularity within all program packages to permit the insertion of additional processing routines with a minimum of effort. In other words, the present system could easily be expanded into a broad base retrieval system which could satisfy the information requirements of a larger set of managers. Recommendations for future modification and expansion of SOIRS are discussed in the following paragraphs.

1. SOIRS should be implemented as soon as possible, thereby eliminating an additional Historical File creation based on BUREAU data which does not contain all of the desired information attributes. Curriculum Officers could effect implementation using the following steps:

a. The collection of data on all students presently attending NPGS.

b. Key punching the subject data.

c. Processing the record additions onto the Current File with the CRDCHK1 and UPDATE programs.

The above steps would be carried out in accordance with section III. D. 2 and Appendix A.

2. Correspondence with the Bureau of Naval Personnel (PERS-N) should be initiated to develop a transaction file which would be used to update the following attributes on an annual basis:

a. Rank
b. Designator
c. Date of rank
d. P-code
e. Retirement year
f. Designator change history
g. Current duty station

After the format of this transaction file has been agreed upon, a program should be written to carry out the subject update. The most complex area of the program would be the insertion of a new duty station, which is easily accomplished because all duty station name fields are of equal length. By moving the logical record into a core position of length 248 bytes and shifting all station names right 16 bytes, the first name field is available for a new current duty station. The right most 16 bytes of this core storage area should not be moved to the updated file.

3. After implementation of SOIRS, user surveys should be conducted to determine if changes to the system are required. Currently 112 bytes of the logical record are reserved for past duty stations, however these areas are blank since the data was not available in ADP form at the BUREAU. Perhaps a requirement for only two past duty stations could be justified, releasing 80 bytes of the record for additional data of a personnel accounting nature such as section assignment and transfer between curricula. The former being an additional

administrative assist for all Curriculum Officers, and the latter being particularly useful to the Engineering Science Curriculum Officer since a considerable number of Engineering Science students do transfer to other curricula areas.

4. Consideration should be given to increasing the size of the SOIRS logical record and inserting data which would support information attributes of the following nature:

a. Refresher course requirements
b. Military housing requirements
c. Text book requirements
d. Classroom scheduling
e. Flight scheduling

These and other attributes could be useful in preparing resource load analysis. For example, the availability of projected class schedules, curricula schedules, and instructor assignments (all derived from a data base) would enhance certain areas of faculty administration, and promote the coordinated development of each separate curriculum. Additional application programs or subroutines would be required to cope with the expanded system. If future changes to SOIRS are not required or are not feasible, the potential of the data base should be exploited through the use of special programs which would replace certain present manual processing chores (e.g., the quarterly computation of OPR is currently accomplished by use of EAM methods).

5. Investigation should be conducted, after sufficient file usage data is accumulated, to determine the feasibility of placing the system into a real time environment. This could be accomplished by placing application programs on the 2314 magnetic disk drives in load

module form, and supplying input data at an on-line terminal. If real time is desired, the most current copies of the files should also be disk resident.

6. Although far removed from actual machine language, list processing languages offer certain advantages over AL. Specifically, these languages by their nature are most effective in an information retrieval application, and should perhaps be considered for SOIRS at a future time.

7. There are two minor problem areas of SOIRS which require additional consideration subsequent to implementation:

a. It is not uncommon to encounter a student who has attended NPGS on two separate occassions. This situation will result in a double entry in the Historical File after completion of a students second period of attendance at NPGS. This is an undesirable condition and can be corrected by establishing an additional education data field within the logical record. If this is done, it might be advantageous to record the undergraduate education of all officers in this additional field.

b. The printed output, resulting from execution of the UPDATE program, currently contains no descriptive labeling. This condition should be corrected if UPDATE is to be used by an individual other than a relatively experienced programmer.

APPENDIX A

NPGS
STUDENT OFFICER INFORMATION
RETRIEVAL SYSTEM
(SOIRS)
USER'S MANUAL

This manual provides a basic introduction to the use of the NPGS Student Officer Information Retrieval System (SOIRS). It is written for those individuals who have not had previous automated data retrieval system experience.

Examples are given throughout the manual to emphasize the basic features of SOIRS. A complete and detailed description of SOIRS may be found in a thesis on a Proposed Student Officer Retrieval System by LCDR's R. L. HENRY and L. L. MASSA.

The information contained in this manual should enable a reader to use SOIRS effectively within a short period of time.

TABLE OF CONTENTS

PART

LIST OF FIGURES

PART 1. INTRODUCTION

The NPGS Student Officer Information Retrieval System SOIRS) consists of four basic parts:

1. Card deck input error checking.
2. BUPERS Historical File initialization.
3. Current File update, and record transfer to Historical File.
4. Current and Historical File information retrieval.

Of these parts, only parts one, three, and four, will be of pertinent interest to the user.

This manual is organized into sections as follows:

1. INTRODUCTION
2. SYSTEM ORGANIZATION
3. PUNCHED CARD RECORD INPUT FORMAT
4. FILE UPDATE AND TRANSFER PROCEDURES
5. FILE SEARCH TECHNIQUES

Sections three, four, and five contain detailed instructions with examples of input/output data including error message's, their possible cause, and correction procedures.

This manual contains the information needed by curriculum officers to implement and use SOIRS effectively but, it is emphasized that it does not provide the information needed for SOIRS file maintenance. The Application Programmer assigned to maintain SOIRS will need the additional information contained in a thesis on a Proposed Student Officer Retrieval System by LCDR's R. L. HENRY and L. L. MASSA on file construction, record format and Job Control Language.

PART 2. SYSTEM ORGANIZATION

SOIRS data is contained on two magnetic tape reels designated the "Current File" and "Historical File". The Current File contains the following data items on all officer students presently attached to the Naval Postgraduate School at Monterey.

1. Name
2. Rank
3. File Number
4. Branch of Service or Foreign Country
5. Sex
6. "IGEP" Indicator
7. Social Security Number
8. Date of Birth
9. Current Designator and Previous Designator with Date of Change if applicable.
10. Date of Rank
11. Original Source Code
12. Year Group
13. Year first eligible to Retire
14. P-Code
15. Quarter and Year started school
16. Expected Quarter and Year of Graduation
17. School Curriculum
18. Expected Level of Education to be achieved (M.S., B.S., etc.)
19. Expected Area in which Degree will be awarded (E.E., O.R., etc.)
20. Current, Graduate and Total "OPR"
21. Chronological list of last eight Duty Stations.

The information contained in data items 3, 6, 9, 11, 12, 13, 14, and 21 is not included for foreign and non-Navy officers and data item 7 is additionally omitted for foreign officers. The Historical File contains identical information on all officer students who have

completed a course of study at NPGS Monterey with actual and final data substituted for the expected and current values contained in the Current File.

Current File initialization is accomplished by locally prepared punched three card deck input for each presently enrolled student officer. Initialization of the Historical File requires a magnetic tape input from the Bureau of Naval Personnel and therefore will only include data on U.S.N. officers currently on active duty. It is expected that when SOIRS is actually initiated by PGS Monterey the Historical File input from BuPers will be augmented with local records available in the Registrar's office to include previous officer students for foreign countries and other U.S. military services.

Current File updating (additions, deletions, and data record changes) and transfer of individual student officer records to the Historical File is accomplished locally using punched card input. Historical File updating can be accomplished from BuPers tape records for applicable data items (Rank, Duty Stations, etc.). It should be noted that all locally prepared three card individual student data decks are error checked by a SOIRS program prior to inclusion on a magnetic tape file.

Current or Historical File data may be retrieved by searching on various combinations of fourteen data record parameters. The search combinations are made up of equal, not-equal and inclusive numeric range comparisons. Each record extracted as a result of a Current or Historical File search is printed and the printed output contains all the information available on the specified file for the selected individual record.

All SOIRS data manipulation programs are coded in Assembly Language and are disk resident in an assembled load module form.

PART 3. PUNCHED CARD RECORD INPUT FORMAT

This part of the SOIRS user manual describes the procedures for making up input data card decks and eliminating data input errors by verifying sixteen key data fields. All three card input data decks, used for Current File initialization or as a Current File record addition with the UPDATE program must have the following individual student officer information keypunched in the columns indicated. A * appearing after the card column numbers signifies that the field is to be left blank for all foreign and non-Navy officers. A ** signifies that the field is to be left blank for foreign officers only.

CARD 1

Column(s)		Information
1 - 6	*	File Number
7 - 36		Last Name(b)first name(b)middle name(b)(b).....(b) (b) = one space, truncate letters in excess of thirty characters.
37		* : If record is to be added to Current Fi
38		Rank Code: from Table "A", Figure 9
39 - 42	*	Designator
43 - 44	*	Year Group
45	*	Year Group subdivision (blank if N.A.)
46 - 48	*	Original Source Code: As listed in the current "Officer Register"
49 - 54		Date of Rank Col. 49, 50 - Year 51, 52 - Month 53, 54 - Day
55 - 58	*	P-Code (blank if N.A.)

CARD 1 (continued)

Column(s)		Information
59 - 64		Date of Birth (Year; Month; Day)
65		Sex (blank if male)
66 - 74	**	Social Security Number
75 - 76	*	Year first eligible to Retire (blank if N.A.)
77 - 79		Curriculum Number
80		Expected level of education at graduation code: From Table "B", Figure 9

CARD 2

Column(s)		Information
1 - 4	*	Date of Designator change (blank if N.A.) (Year; Month)
5 - 8	*	Old Designator (blank if N.A.)
9		School start quarter
10 - 11		School start year
12		Expected graduation quarter
13 - 14		Expected graduation year
15 - 17		Current graduate OPR
18 - 20		Current total OPR
21		Expected Degree at graduation code: From Table "C", Figure 9
22 - 23		Branch of Service or Foreign Country: From Table "D", Figure 9
24		Number of past Duty Stations (including NPGS, Monterey)
		Note: U.S.N. Officers must have a minimum of 1 and a maximum of 8 in this column. All non-USN students must have a "0" in this column.
25 - 40	*	S(b)PG(b)Monterey(b)(b)(b) (b) = one blank space

CARD 2 (continued)

Column(s)		Information
41 - 56	*	Previous Duty Station (blank if N.A.)
57 - 72	*	Second previous Duty Station (blank if N.A.)
73 - 80		Blank

CARD 3

Column(s)		Information
1 - 16	*	Third previous Duty Station (blank if N.A.)
17 - 32	*	Fourth previous Duty Station (blank if N.A.)
33 - 48	*	Fifth previous Duty Station (blank if N.A.)
49 - 64	*	Sixth previous Duty Station (blank if N.A.)
65 - 80	*	Seventh previous Duty Station (blank if N.A.)

Prior to inclusion of new student officer data decks on Current File magnetic tape they must be processed by the SOIRS input card deck verification program to eliminate errors in key data record fields. Sixteen input card data fields are checked for errors. If any errors are detected an error message is printed that specifies the incorrect data field and prints the non-valid character(s) found. A list of sixteen possible error messages, with the indicators * or ** used in lieu of non-valid characters, is shown below:

1. WARNING CARD DECK DELETED DUE TO ERRONEOUS DATA.
2. INCORRECT AREA CODE = * .
3. INCORRECT CURR CODE = * .
4. INCORRECT CTY CODE = ** .
5. INCORRECT DEGR CODE = * .
6. INCORRECT DESG CODE = * .
7. INCORRECT DUTY CODE = * .
8. INCORRECT EDTE CODE = * .
9. INCORRECT GOPR CODE = * .

10. INCORRECT RANK CODE = * .

11. INCORRECT SDTE CODE = * .

12. INCORRECT SRNR CODE = * .

13. INCORRECT SSNR CODE = * .

14. INCORRECT SVC CODE = ** .

15. INCORRECT TOPR CODE = * .

16. NEW RECORD IND MISSING .

A description of the type of error checking performed, on each of the sixteen data fields tested, with an example of each resulting error message from the sample output contained in Figure 7, is shown below:

1. NAME FIELD: First ten columns checked for non-blank and alphabetic characters. Result of error(s) = card deck rejected and no printed output except the statement, " WARNING CARD DECK DELETED DUE TO ERRONEOUS DATA ". Shown in Figure 7.

2. DEGREE AREA CODE: Checked for valid alphabetic characters " A " through " T ". Result of error = error message printed as follows, "INCORRECT AREA CODE = X ". Shown in Figure 7.

3. CURRICULUM: Checked for non-numeric characters (including blanks). Result of error(s) = incorrect curriculum number and the following error message, " INCORRECT CURR CODE = A ", printed as shown in Figure 7.

4. FOREIGN COUNTRY CODE: Checked for valid numeric codes " 20 " through " 41 ". Result of error = error message printed as follows, " INCORRECT CTY CODE = 51 ". Shown in Figure 7.

5. EDUCATIONAL LEVEL CODE: Checked for non-numeric characters (including blanks). Result of error = error message printed as follows, " INCORRECT DEGR CODE = J ". Shown in Figure 7.

6. DESIGNATOR: Checked for non-numeric characters (including blanks). Result of error(s) = incorrect Designator and the following error message, " INCORRECT DESG CODE = / ", printed as shown in Figure 7.

7. NUMBER OF PAST DUTY STATIONS: Checked for non-numeric characters (including a blank) and maximum numeric value of eight. Result of error = only three duty stations and the following error message, " INCORRECT DUTY CODE = 9 ", printed as shown in Figure 7.

8. GRADUATION DATE: Checked for non-numeric characters (including blanks). Result of error(s) = incorrect Graduation quarter and year and the following error message, " INCORRECT EDTE CODE = / ", printed as shown in Figure 7.

9. GRADUATE "OPR": Checked for non-numeric characters (including blanks). Result of error(s) = incorrect Graduate OPR and the following error message, " INCORRECT GOPR CODE = B ", printed as shown in Figure 7.

10. RANK CODE: Checked for valid alphabetic characters " A " through " S ". Result of error = error message printed as follows, " INCORRECT RANK CODE = & ". Shown in Figure 7.

11. SCHOOL START DATE: Checked for non-numeric characters (including blanks). Result of error(s) = incorrect school start quarter and year and the following error message, " INCORRECT SDTE CODE + L ", printed as shown in Figure 7.

12. SERVICE NUMBER: Checked for non-numeric characters (including blanks). Result of error(s) = incorrect Service Number and the following error message, " INCORRECT SRNR CODE = I ", printed as shown in Figure 7.

13. SOCIAL SECURITY NUMBER: Checked for non-numeric characters (including blanks). Result of error(s) = incorrect Social Security Number and the following error message, " INCORRECT SSNR CODE = U ", printed as shown in Figure 7.

14. BRANCH OF SERVICE CODE: Checked for the seven printed valid numeric codes 00, 01, 02, 10, 11, 12, 13. Result of error = error message printed as follows, " INCORRECT SVC CODE = 04 ". Shown in Figure 7.

15. TOTAL OPR: Checked for non-numeric characters (including blanks). Result of error(s) = incorrect Total OPR and the following error messages, " INCORRECT TOPR CODE = I ". Shown in Figure 7.

16. NEW DATA DECK INDICATOR: The following statement will be printed if indicator is not present. " NEW RECORD IND MISSING ". Shown in Figure 7.

Figure 7 is sample output from the input card verification program and contains examples of each error message and statement described above. Additional entries are listed with multiple error messages because error messages will be generated for all errors found in each data card deck.

Figure 8, illustrates the result of correcting the errors contained in Figure 7 and re-verifying the input card decks. It is emphasized that an error free output as shown in this figure is mandatory before the input card decks can be incorporated in the Current File.

Figure 9 is a recommended hand-out for incoming student officers to use when providing curriculum officers with the data needed for initialization and updating SOIRS Current File. The hand-out contains

all required tables with complete and specific instructions for filling out the attached form which can be easily used by keypunching personnel in preparing the individual officer's three card data input deck.

INPUT CARD VERIFICATION PROGRAM OUTPUT LISTING WITH ERRORS

NAME RANK, FILE NR, SVC/CTY(,*) (*=F IF FEMALE,=I IF IGEP)	SOCIAL SECURITY NUMBER	BIRTH DATE YR-M-D	DESIG (CHG)	RANK DATE YR-M-D	ORIG SOURCE CODE	YR GP	1ST RET YR	P CODE	NPGS QTR-YR ST/END	CUR PIC LUM	DEGR / AREA	OPR G/I	PAST DUTY STAS
ACADEMY JOE FRESHCAUGHT ENS ,609912,USN	* 109453867	481225	1100	680603	01-0	60	83		1-68 1-69	380	MS MATS	259 241	USNA S PG MONTEREY
AIRFORCE CHARLY FLYTYBIRD **** INCORRECT RANK CODE= 6 **** USAF	* 689469012	361010		651111					2-54 2-56	610	BACH A.E.	200 175	
AVERS MAXIMUM MOST **** INCORRECT RANR CODE= I **** LCDR,631892,USN	* 631892123	321206	1100	661225	01-0	58	79	8801	3-67 2-69	427	MS C.S.	278 276	S PG MONTEREY USS MOONFISH USS STARFISH USS TUNAFISH USS PINEFISH
ARMYTYPE ALPHONSE KNOTHEAD GEN , USA	* 689468012	361010		651111					3-65 4-66	610	BACH NONE	125 139	
**** INCORRECT DEGR CODE= J **** BADWIN RONALDO SQUARE LT ,269052,USN	* 269052123	390309	1400 (6901) (1109)	690226	07-0	64			3-67 2-69	360	AENG	285 290	S PG MONTEREY MACV USS AMERICA
********************	WARNING CARD DECK DELETED DUE TO ERRONEOUS DATA								********************				
**** INCORRECT SSNR CODE= U **** CIVIL SERVICE CONOTHING CIV , USNR	* UUUUUUUUU	420404							2-69 4-70	530	PG NONE	050 110	
DOWNUNDER WAY DOWN GO ENS ,561234,USNR	* 234567891	490707	1300	650404	01-0	68			2-67 3-69	817	3YRC AENG	300 300	STA NUMBER ONE STA NUMBER TWO STA NUMBER THREE STA NUMBER FOUR STA NUMBER FIVE STA NUMBER SIX STA NUMBER SEVEN
**** INCORRECT CURA CODE= A **** DUMMY DUMMER DUMMY LTJG,654311,USN	* 987654321	290303	1100 (5903) (1110)	541111	05-0	40			5-41 4-69	A80	PO A.E.	050 050	STA NUMBER ONE STA NUMBER TWO STA NUMBER THREE STA NUMBER FOUR STA NUMBER FIVE STA NUMBER SIX STA NUMBER SEVEN STA NUMBER EIGHT
FLEAGLE OOGLE EYE **** INCORRECT AREA CODE= X **** LT ,123456,USNR	* 123456789	390809	1510	671212	21-1	60			1-68 1-70	610	BACH	123 123	S PG MONTEREY USS BIGCITY USNSY PORTSMOUTH
**** INCORRECT DESG CODE= / **** GEORGE LZERO, LOOP **** INCORRECT SVC CODE=04 **** LT ,531320,	* 531320123	390309	13// (6902) (1310)	661215	07-0	61	89	8601	2-68 3-70	514	CERT MGT.	180 185	S PG MONTEREY NAS ALAMEDA R MON-5
**** INCORRECT GOPR CODE= 5 **** HATCH WATER TIGHT CDR ,360497,USN	* 896-35432	330505	1100 (6406) (1109)	690402	07-0	55	74		3-68 3-70	427	MS C.E.	839 205	S PG MONTEREY USS DIRTYCAN CRUDESLANT USS LEAKYCAN USS C...E COUNTY SERVEPAC MSC-26 MSC-62

FIGURE 7

```
NAME
RANK,FILE NR,SVC/CTY(,*)          SOCIAL     BIRTH          RANK     ORIG                                NPGS    CUR   DEGR
1*** IF FEMALE,*1 IF IGEP)        SECURITY   DATE   DESIG   DATE     SOURCE   YR   1ST    P     QTR-YR   RIC   /     OPR
                                  NUMBER     YR-M-D (CHG)   YR-M-D   CODE     GP   RET YR CODE  ST/END   LUM   AREA  G/Y  PAST DUTY STAS
HENRY ROBERT LEE
**** INCORRECT TOPR CODE= I ****  * 549426211  360717  1400  671010  07-0  59  79        1-67  360  MS    240  NPGS MTY
LCDR,AT1094,USN                                     (6901)                                2-69        AENG  212  DD-728
                                                    (1100)                                                       CMSF SFRAN
                                                                                                                 DER-389
**** INCORRECT SOTE CODE= L ****
HOARY ROBERT HEAVYSET             * 649231123  360603  1100  681212 . 07-0  60  80        3-67  360  MS    232  S PG MONTEREY
LCDR,649231,USN                                     (4406)                                2-69        AENG  235  USS SLOWMOVER
                                                    (1105)                                                       12NDHQ
                                                                                                                 USS ROLYPCLV
*******************          WARNING CARD DECK DELETED DUE TO ERRONEOUS DATA          ********************
KILOWATT VERY READY
**** INCORRECT EDTE CODE= / ****  * 224466031  470606  1100  690601  01-0  68      8501  1-69  535  MS    242  S PG MONTEREY
ENS ,246913,USN                 I                                                         1-7/        E.F.  263  USNA ANNAPOLIS
LACY JOHNATHON PORSCHE              600992123  390309  1100  690601  01-0  60  79        3-67  360  MS    219  S PG MONTEREY
LCDR,600992,USN                                                                           2-69        AENG  228  USS NUCKYPOO
**** NEW RECORD IND MISSING ****
                                                                                                                 USS DIESLTYPE
**** INCORRECT DUTY CODE= 9 ****                                                                                 USNA ANNAPOLIS
LOCKER DAVY JONES
CDR ,444111,USN                   * 679367541  320306  1100  680204  01-0  54  73        3-68  426  MS    205  S PG MONTEREY
                                                                                          3-70        NATS  239  USS CANNYCAN
                                                                                                                 CRUDESPAC
LOCKSTEP PUNCHLY RICHARD          * 331155779  420731  1105  681215  07-0  66            1-67  517  PHD   276 .S PG MONTEREY
LTJG,331155,USNR                                                                          4-69        MGT.  288

**** INCORRECT DUTY CODE= K ****
**** INCORRECT SSNR CODE= K ****
**** INCORRECT SOTE CODE= O ****
**** INCORRECT CURR CODE= O ****
**** INCORRECT DEGR CODE= W ****
**** INCORRECT GOPR CODE= I ****
MASSA LAWRENCE LEE                  KJK377KDI  340909  1400  670601  21-1  58  70        O-L   060         IO.  NPGS MTY
**** INCORRECT RANK CODE= W ****
**** INCORRECT SRNR CODE= P ****
**** INCORRECT SVC  CODE=09 ****
**** INCORRECT EDTE CODE= I ****
**** INCORRECT AREA CODE= * ****
**** INCORRECT TOPR CODE= T ****
     62P744,
**** NEW RECORD IND MISSING ****                    (6901)                                I-L.              IUJ  LST-1150
                                                    (1100)                                                       LST-1176
MCGILLACUDDY FLIRTEYES SALLY    F * 345678912  480505  1100  680505  07-0  67  87  1234  1-66  360  PHD   295  STA NUMBER ONE
LTJG,456123,USN                                                                           1-70        ESCI  283  STA NUMBER TWO
                                                                                                                 STA NUMBER THREE
                                                                                                                 STA NUMBER FOUR
                                                                                                                 STA NUMBER FIVE
                                                                                                                 STA NUMBER SIX
**** INCORRECT SSNR CODE= L ****
**** INCORRECT DEGR CODE= I ****
MORSA LORENZO LEAN
**** INCORRECT AREA CODE= X ****  * LI,MI,123  340908  1100  660601  21-1  58  75        3-67  360        212  S PG MONTEREY.
LCDR,628728,USN                                                                           2-69              214  USS FLATBOTTEM
                                                                                                                 USS RUSTBUCKET
                                                                                                                 USS FIRSTRIDE
```

INPUT CARD VERIFICATION PROGRAM OUTPUT LISTING WITHOUT ERRORS

NAME RANK, FILE NR, SVC/CTY(*) [illegible]	SOCIAL SECURITY NUMBER	BIRTH DATE YR-M-D	ORIG [illegible]	RANK DATE YR-M-D	ORIG SOURCE CODE	YR GP	1ST RET YR	P CODE	NPGS QTR-YR ST/END	CUR RIC LUM	DEGR / AREA	OPR SZI	PAST DUTY STAS
ACADEMY JOE FRESHCAUGHT ENS ,[illegible]12,USN	* 1 [illegible]3867	481225	11[illegible]	481023	11-[illegible]	[illegible]	88		1-68 1-69	380	MS MATS	259 241	USNA S PG MONTEREY
AIRFORCE CHARLY FLYTYBIRD MAJ , USAF	[illegible]12	361.1		651111					2-54 2-56	610	BACH A.E.	220 175	
AKERS MAXIMUM MOST LCDR,43[illegible]551,USN	* [illegible]2123	3212[illegible]	11 [illegible]	561225	[illegible]1-[illegible]	58	79	8801	3-67 2-69	427	MS C.S.	278 276	S PG MONTEREY USS MOONFISH USS STARFISH USS TUNAFISH USS PINEFISH
ARMYTYPE ALPHONSE KNOTHEAD GEN , USA	* [illegible]12	361 1		651111					3-65 4-65	610	BACH NONE	125 139	
BACWIN RONALDO SQUARE LT ,269[illegible]92,USN	* 26[illegible] [illegible]2123	3[illegible]	14[illegible] ([illegible]) (11[illegible])	59 226	[illegible]7-[illegible]	64			3-67 2-69	360	MS AENG	285 290	S PG MONTEREY MACV USS AMERICA
BARNALD ROBERTO SKINHEAD LT ,723614,USN	* 723[illegible]4123	35[illegible]	1320	671215	07-[illegible]	62	81		3-67 2-69	360	MS AENG	225 232	S PG MONTEREY NAS WHIDBY NAS ALAMEDA
CIVIL SERVICE CONDTHING CIV , USNR	* 111111111	42[illegible]							2-69 4-70	530	PG NONE	050 110	
DOWNUNDER WAY DOWN GC ENS ,561234,USNR	* 234[illegible]7891	4[illegible]7	13[illegible]	6[illegible]404	[illegible]1-[illegible]	68			2-67 3-69	817	3YRC AENG	300 300	STA NUMBER ONE STA NUMBER TWO STA NUMBER THREE STA NUMBER FOUR STA NUMBER FIVE STA NUMBER SIX STA NUMBER SEVEN
DUMMY DUMMER DUMMY LTJG,[illegible]4311,USN	* [illegible]7654321	2[illegible]	11[illegible] ([illegible]) (11 [illegible])	541111	[illegible]5-[illegible]	4[illegible]			4-61 4-69	380	PG A.E.	050 050	STA NUMBER ONE STA NUMBER TWO STA NUMBER THREE STA NUMBER FOUR STA NUMBER FIVE STA NUMBER SIX STA NUMBER SEVEN STA NUMBER EIGHT
FLEAGLE COOLE [illegible] LT ,123456,USN	* 123456789	35[illegible]	151[illegible]	571212	21-1	6[illegible]			1-68 1-70	610	BACH NONE	123 123	S PG MONTEREY USS BIGCITY USNSY PORTSMOUTH
GEORGE LEEROY LOOP LT ,[illegible]3123[illegible],USN	* [illegible]3132 123	39[illegible]	13[illegible] ([illegible]2) (131[illegible])	661215	[illegible]7-[illegible]	61	89	8601	2-68 3-70	514	CERT MGT.	190 185	S PG MONTEREY NAS ALAMEDA TRARON-5
HATCH WATER TIGHT CDR ,3[illegible]497,USN	* [illegible]35432	33[illegible]	11 [illegible] ([illegible]) (11[illegible])	5[illegible]402	[illegible]7-[illegible]	[illegible]5	74		3-68 3-70	427	MS C.E.	239 205	S PG MONTEREY USS DIRTYDAN CRUDESLANT USS LEAKYCAN USS CLOSE COUNTY SERVEPAC MSO-26 MSC-62
HENRY ROBERT LEE LCDR,431[illegible]4,USN	* [illegible]4211	3[illegible]717	1[illegible] ([illegible]1) (11 [illegible])	571[illegible]	[illegible]7-[illegible]	59	79		1-67 2-69	360	MS AENG	240 222	NPGS MTY DD-728 CWSF SFRAN DER-389

FIGURE 8

FIGURE 9

INSTRUCTIONS TO INCOMING STUDENTS FOR COMPLETING NPGS

STUDENT OFFICER INFORMATION RETRIEVAL SYSTEMS (SOIRS) DATA FORM

INCOMING STUDENT OFFICER

CURRICULUM OFFICE DATA SHEET

The attached data sheet is divided into three IBM card images. The data sheet is to be filled in exactly as instructed below. Do not use lower case letters or substitute blanks for zeros. Each data field block must be completely filled in unless specifically stated otherwise. Foreign officers should not fill in items marked with * or **, non-Navy U.S. officers should not fill in items marked with *.

CARD 1

Block(s)		Required Data
1 - 6	*	File Number
7 - 36		Last name (b) first name (b) middle name (b)(b)...(b) (b) indicates blank spaces Omit portion of name that exceeds thirty letters.
38		Rank Code: From Table " A ".
39 - 42	*	Designator
43 - 44	*	Year Group
45	*	Year Group Subdivision (blank if N.A.).
46 - 48	*	Original Source Code: As listed in "Officers Register".
49 - 54		Date of Rank (Year, Month, Day).
55 - 58	*	P-Code (blank if N.A.).
65		Sex: Insert "F" if Female, Male officers leave blank.
66 - 74	**	Social Security Number
75 - 76	**	Year first eligible to Retire (blank if N.A.).
77 - 79		Curriculum Number
80		Expected Level of Education at Graduation Code: From Table "B".

CARD 2

Block(s)		Required Data
1 -	*	Designator Change Data: Year and Month (blank if N.A.).
5 - 8	*	Old Designator (blank if N.A.).
9 - 11		School Start Date: Quarter and Year.
12 - 14		Expected Graduation Date: Quarter and Year.
15 - 17		Current Graduate QPR; 000 if none established.
18 - 20		Current Total QPR: 000 if none established.
21		Expected Degree at Graduation Code: From Table "C".
22 - 23		Branch of Service or Foreign Country Code: From Table "D".
24		Number of Duty Stations listed below. Note: USN officers must have a minimum of one and a maximum of eight in this block. All non-USN officer students must have a "0" in this column.
41 - 56	*	Previous Duty Station (blank if N.A.).
57 - 72	*	Second previous Duty Station (blank if N.A.).

CARD 3

Block(s)		Required Data
1 - 16	*	Third previous Duty Station (blank if N.A.)
17 - 32	*	Fourth previous Duty Station (blank if N.A.)
33 - 48	*	Fifth previous Duty Station (blank if N.A.)
49 - 64	*	Sixth previous Duty Station (blank if N.A.)
65 - 80	*	Seventh previous Duty Station (blank if N.A.)

TABLES

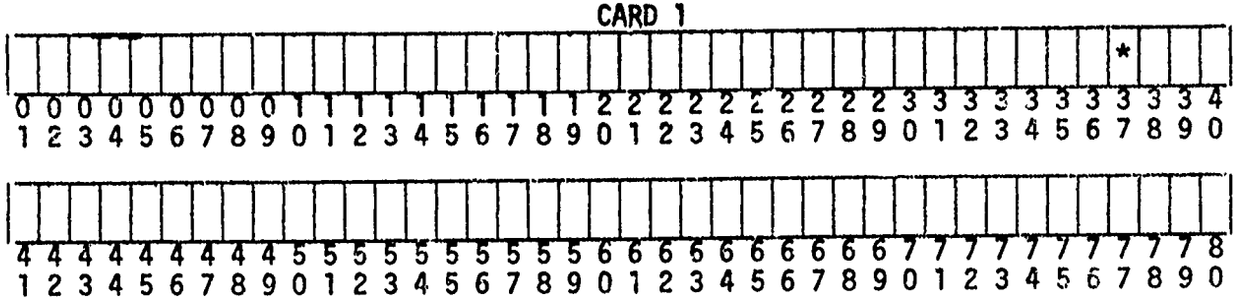
"A"	
FADM	A
ADM	B
VADM	C
RADM(U)	D
RADM(L)	E
COMO	F
CAPT	G
CDR	H
LCDR	I
LT	J
LTJG	K
ENS	L
CIVILIAN	M
2LT	N
1LT	O
MAJ	P
LCOL	Q
COL	R
GEN	S

"B"	
LS THN HS	0
HIGH SCHL	1
LS 2Y COL	2
2 YRS COL	3
3 YRS COL	4
PG NO DGR	5
BACHELOR	6
CERTIFICATE	7
MASTERS	8
DOCTORS	9

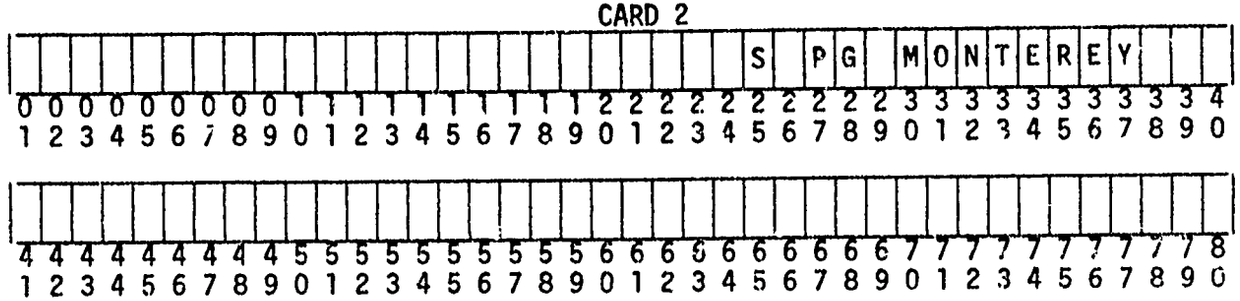
"C"	
NONE	A
A.E	B
AERO ENG	C
ENG SCI	D
INT REL	E
E.E	F
ELEC ENG	G
METEROLOGY	H
OCEANOGRAPHY	I
O.R	J
COMP MGT	K
COMP SCI	L
MGT	M
M.E	N
MECH ENG	O
MATH	P
PHYSICS	Q
CHEM	R
MAT SCI	S
C.E	T

"D"	
USN	00
USN(IGEP)	01
USNR	02
USA	10
USCG	11
USMC	12
USAF	13
BRAZIL	20
CANADA	21
CEYLON	22
CHILE	23
CHINA	24
COLUMBIA	25
EQUADOR	26
GERMANY	27
INDIA	28
IRAN	29
ISRAEL	30
KOREA	31
NORWAY	32
PERU	33
PHILIPPINES	34
PORTUGAL	35
THAILAND	36
TURKEY	37
URUGUAY	38
VENEZUELA	39
VIETNAM	40
OTHER	41

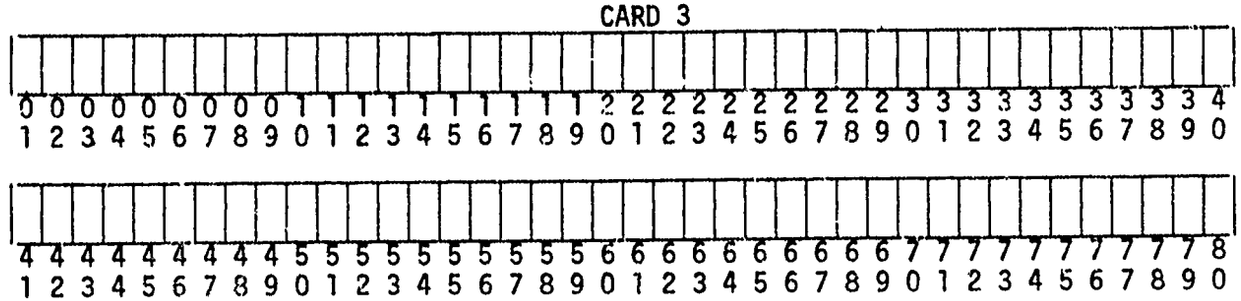
CARD 1



CARD 2



CARD 3



PART 1. FILE UPDATE AND TRANSFER PROCEDURE

This part of the SOIRS user manual describes the procedures for updating the Current File (including additions and deletions) and transferring departed students records from the Current to the Historical File.

SOIRS file updating is accomplished by the Current File Update and Historical File Transfer Program (UPDATE) in one step. This program adds, deletes, and changes Current File records in addition to transferring records to the Historical File (records may be updated during the transfer process). The following student officer record data items can be changed using the UPDATE program:

1. Designator (including date of change)
2. Rank
3. Date of Rank
4. Curriculum
5. Degree Area
6. Educational Level
7. Graduate and Total QPR
8. Graduation Date (Quarter and Year)

The punched card deck format for records to be added to the Current File is described in detail in Part 3. Record deletion, change, and transfer requires a single punched input card which is described below. Any data item that is not to be changed may be left blank. The * and NAME fields for foreign officers and Social Security Number field for all other students must be filled in because these fields are used to identify the desired Current File record. Items marked with a * apply to foreign officers only and items marked with ** apply to U.S.N.

officers only.

1. RECORD DELETION CARD

Column(s)		Information
1	*	*: Foreign Officer Indicator
2 - 29	*	Name: Exactly as it appears in Current File tape records
1 - 19		Name: First nineteen characters only
20 - 28		Social Security Number
29 - 79		Blank
80		%: Record Deletion Indicator

2. RECORD UPDATE AND TRANSFER CARDS

Column(s)		Information
1	*	*: Foreign Officer Indicator
2 - 29	*	Name: Exactly as it appears in Current File record
1 - 19		Name: First nineteen characters only
20 - 28		Social Security Number
30 - 33	**	New Designator
34 - 39	**	Date of Designator change: Year and Month
40		Rank Code: From Table "A", Figure 9
41 - 46		Date of Rank: Year, Month and Day
50 - 52		Curriculum
54		Degree Area Code: From Table "C", Figure 9
56		Educational Level: From Table "B", Figure 9
60 - 62		Graduate QPR
63 - 65		Total QPR
70 - 72		Graduation Date: Quarter and Year
80		$: Record transfer indicator, leave blank if update only is desired

Examples of each type of UPDATE input card are contained in Figure 10 and a standard coding sheet should be used to provide keypunch information. It should be noted that all possible data changes are shown in the examples of Figure 10 but they may be left blank if a particular data item is unchanged.

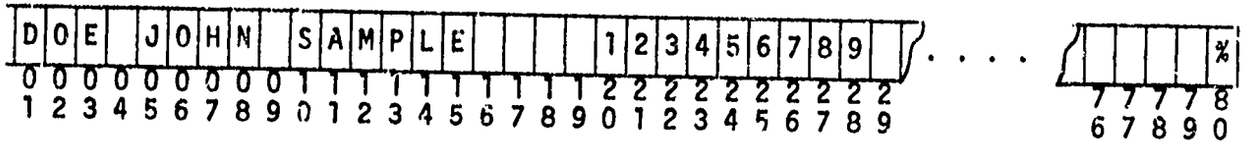
The delete, update and transfer data input cards used in the UPDATE program must be in alphabetical order and precede the three card data input decks that are to be added to the Current File (these do not have to be in alphabetical order), although each three card deck must be in correct order.

In addition to effecting the required Current and Historical File magnetic tape record changes the UPDATE program provides printed output to verify each data record affected. If a delete, update, or transfer data input card is not in correct alphabetical sequence or the * and NAME fields for foreign officers and Social Security Number field for all other students is incorrect, an error message will be printed for that input card as follows: " PROGRAM WILL NOT RUN. DATA UPDATE CARD FOR (student name) OUT OF SEQUENCE OR INCORRECT." If one or more error messages occur during execution of the UPDATE program the data cards containing errors must be corrected and the program re-executed for all input data cards. For successful execution of the UPDATE program no error messages may occur.

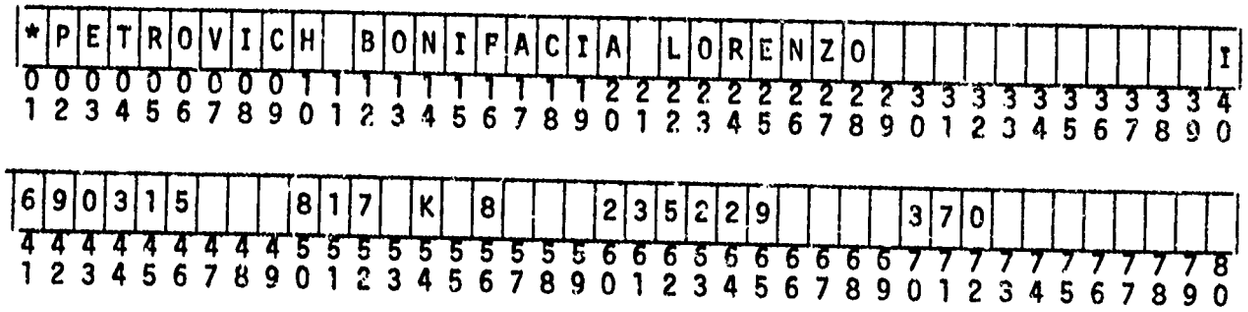
Figure 11 is a sample output listing from the UPDATE program containing several examples of record addition, deletion, updating and transfer. Record addition and deletion produces a single line of printed output containing the first one hundred and thirty-three (133) characters of the affected record with the appropriate indicator,

"* NEW *" or "*DELETE*" appearing at the right edge of the printed line. Record updating and transfer produces two lines of single spaced printed output. The first line contains the first one hundred and thirty-three (133) characters of the original record. The second line contains the same data but includes all changes effected and the appropriate indicator "* UPDATE*" or "* XFER *" at the right edge of the printed line.

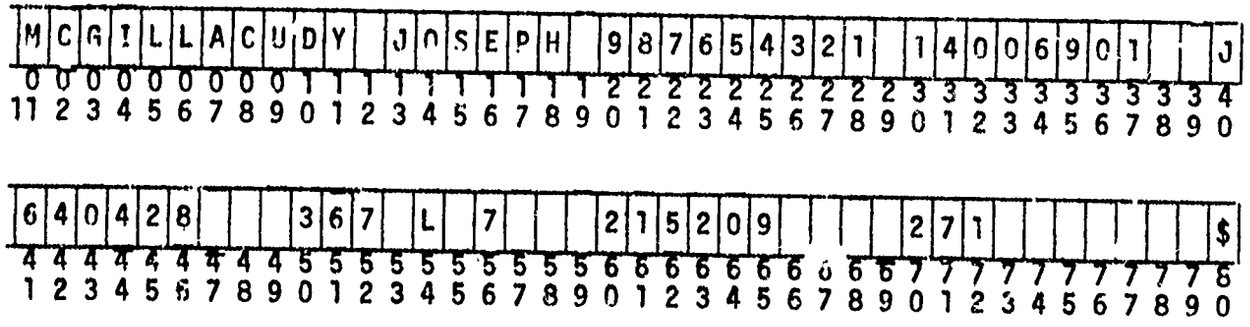
RECORD DELETION INPUT CARD (NON-FOREIGN STUDENT)



RECORD UPDATE INPUT CARD (FOREIGN STUDENT)



RECORD TRANSPER INPUT CARD (NON-FOREIGN STUDENT)



FIGURE

CURRENT FILE UPDATE AND TRANSFER PRINTED OUTPUT LISTING

```
##########        START OF DATA          ##########        START OF DATA          ##########           START OF DATA          #########
608912ACADEMY JOE FRESHCAUGHT            L1100608010680603     481225 109453867883838          168169225924150125USNA              S PG MONTERE
608912ACADEMY JOE FRESHCAUGHT            L1100608010680603     481225 109453867883671          166169233234S012USNA                S* UPDATE *E
      ARMYTYPE ALPHONSE KNCTHEAD          S           651111     361010 689468012  6106          365466125139A100                     * DELETE *
269052BADMIN RONALDO SQUARE              J140064007C690226     393309 269052123  3608690111053672692852900003S PG MONTEREY  MACV
269052BADMIN RONALDO SQUARE              J14006400706902226    390309 269052123  36085901110536726927528000035 PG MONTEREY  P* UPDATE *
654311DUMMY DUMMER DUKMY                 K1100400050541111     290303 987654321  380559C311004614690500508008STA NUMBER ONE  S* DELETE *T
123456FLEAGLE OOGLE EYE                  J151060021167121 2    390809 123456789  6106          16817012312YA023S PG MONTEREY    USS BIGCITY
123456FLEAGLE OOGLE EYE                  J1510600211671212     390809 123456789  6106          1681702222224023S PG MONTEREY    USS* XFER *
369497HATCH WATER TIGHT                  H1120550070690422     333505 89673543274427864061105368370239205T008S PG MONTEREY  USS DIRTYOAN
369497HATCH WATER TIGHT                  H1120550070690402     333505 89673543274427864061105368370245233T008S PG MONTEREY  U* UPDATE *N
649231HOARY ROBERT HEAVYSET              I1100630070681212     360603 649231123803608640611053672692322 39C004S PG MCNTEREY  U* DELETE *E
767676JONES JOHN PAUL                    A130049 010620796     280706 007600761884209          168170250250I008S PG MONTEREY    USS CONSTITU
767676JONES JOHN PAUL                    A130049 010620796     280706 0076007618881790269110016827024222 4M008S PG MCNTEREY  U* UPDATE *U
625748MASSA LAWRENCE LEE                 I140058021167010l     340809M5424055327336086901110036726923921 4C005NPGS MTY   LST-1150
625748MASSA LAWRENCE LEE                 I140058021167010l     340809M54540055327384086901110367269295233C005NPGS MTY   L* UPDATE *
      PERUVIAN GUNFLASH DOMINIC           F           630827     401212            6009          164466234299T330
      PERUVIAN GUNFLASH DOMINIC           F           630827     401212            6009          164466210215T330                     * UPDATE *
      PETERSON PERRY CANNONCOCKER         Q           690316     310607 624800039  9278          368470201202T120
      PETERSON PERRY CANNONCOCKER         Q           690316     310607 624800039  5278          368369212216T120                     * XFER *
869432RESERVE ALSWAYS READY              J110560 07065022683043 70226 998877665  8178          169370210215K024S PG MONTEREY    NRTC GLAKES
869432RESERVE ALSWAYS READY              I110560 07069030183043 70226 998877665  8178          169370225234K024S PG MONTEREY    N* UPDATE *
091362WIGGLEMORE SUZY SWIVELHIPS         K165070 1906812339803460406F811362123  5858          168270225252R022S PG MCNTEREY    NAB CORONADO
691362WIGGLEMORE SUZY SWIVELHIPS         K165070 1506812339803463406F811362123  5858          168270220243R022S PG MONTEREY    N* UPDATE *O
      YING YANG YAREE                     S           651025     300615            6109          268369190216040 0
      YING YANG YAREE                     S           651025     300615            6109          268369185212040 0                    * XFER *
002678LUCKY GO HAPPY                     G130050007066120199993012 01 00469328994814760121310169270229213M006STA NUMBER ONE  STA * NEW *
      AIRFORCE CHARLY FLYTYBIRD           P           651111     361010 689468012  6106          254256200175B130                       * NEW *
530581AKERS MAXIMUM POST                 I110058 01066225880132 1206 631892123794278          367269278276L005S PG MCNTEREY    USS * NEW *
      SAILOR SHALLOW WATER                I           670810     301018 862930467  5258          268370215224011 0                      * NEW *
824428PROTALONKA MICHELE RUMSOT          J110565007069050 1     400430 824428123  3608          367269280285C022S PG MCNTEREY    USS * NEW *
825666FESAR JOHN ARNOLD                  J14056002116809198362361222 825666666826109          467170213232C006S PG MCNTEREY    USS * NEW *
456123MCGILLACUDDY FLIRTEYES SALLY       K110067 0706805051234483505F345678912873609          166170295283D006STA NUMBER ONE   STA * NEW *
628728MODSA LORENZO LEAN                 I110058021166060 1     340403 628728123755608          367269212214C004S PG MCNTEREY    USS * NEW *
331155LOCKSTEP PUNCHLY RICHARD           K110566 C70681215     420731 331155779  5179          167469276288M021S PG MCNTEREY      * NEW *
444111LOCKER DAVY JONES                  H110054010680204      320606 679367541734268          368370205239S008S PG MCNTEREY    USS * NEW *
600992LACY JOHNATHON PORSCHE             I1100600010690601     390309 600992123793608          367269219228C004S PG MCNTEREY    USS * NEW *
246813KILOWATT VERY READY                L110068 010680601850147 0606 224466881  5358          169170242263F012S PG MCNTEREY    USNA* NEW *
631894HENRY ROBERT LEE                   I140359007067131 0     360717M549426211793608690111001672692402220004NPGS MTY   DD-7* NEW *
531320GEORGE LEERCY LOOP                 J130061 07066121556013 90309 5313201236951476902131026837018018 5M003S PG MCNTEREY  NAS * NEW *
723614BARNALD ROBERTO SKINHEAD           J132062007067121 5     350503 723614123813608          367269225232C003S PG MCNTEREY    NAS * NEW *
##########        END OF DATA            ##########        END OF DATA            ##########           END OF DATA            #########
```

FIGURE 11

PART 5. FILE SEARCH TECHNIQUES

This part of the SOIRS user manual describes the procedues for extracting required records from the Current or Historical File. Current and Historical File record retrieval is effected with one to twenty, free-form, punched input data cards containing valid combinations (each key used must be followed by at least one space) of fourteen search parameter keys. The Application Programmer must be told which file is to be searched as the keys are common to both the Current and Historical Files. These types of searches, EQUAL, NON-EQUAL or GREATER-THAN/LESS-THAN, may be conducted either singly or in combination. Each search compares all individual records in the specified file with the data specified on the input data card(s) and extracts only those records that are respectively equal, not-equal, or within the specified numeric range. Permissible record data field search parameters and their keys are listed below:

DATA FIELD SEARCH PARAMETER	KEY	PERMISSIBLE SEARCH TYPE(S)
1. Social Security Number	A	Equal
2. Sex	B	Equal
3. P-Code	C	Equal Not-equal
4. Designator	D	Equal Not-equal
5. Branch of Service or Foreign Country Code (from Table "D", Fig.9)	E	Equal Not-equal

NOTE: A secondary key of "N" for all USN officers, "C" for all Foreign officers and "U" for all non-Navy U.S. officers may be used in addition to the primary key. (E=N, E#E, E=U, etc.)

DATA FIELD SEARCH PARAMETER	KEY	PERMISSIBLE SEARCH TYPE(S)
6. Year of Birth	F	Equal Not-equal Greater-than/Less-than
7. Year Group	G	Equal Not-equal Greater-than/Less-than
8. Rank Code (from Table "A", Fig. 9)	H	Equal
9. Curriculum	I	Equal
10. Degree Area (from Table "C", Fig. 9)	J	Equal
11. Educational Level	K	Equal
12. Graduate QPR (To nearest 10th)	L	Equal
13. Total QPR (To nearest 10th)	M	Equal
14. Graduation Year	N	Equal Greater-than/Less-than

The character " = " follows the search parameter key when an " EQUAL " search is desired. An equal search will test the specified data field of each individual record contained in the designated file (Current or Historical) for an exact match with the character(s) following the equal search indicator and print the complete record if a match is found. For example, if a list of all previously graduated students who were enrolled in curriculum 367 was wanted, the Historical File would be searched using a data card containing I = 367 as the search parameter key.

The character " # " follows the search parameter key when a " NOT-EQUAL " search is desired. A not-equal search will test the specified data field of each individual record contained in the designated field

for an exact match with the character(s) following the not-equal search indicator and print the complete record if they do not match. For example, if a list of all USNR students presently enrolled in NPGS was wanted, the Current File would be searched using a data card containing E#02 as the search parameter key.

The character " % " follows the search parameter key when a " GREATER-THAN/LESS-THAN " search is desired (a comma must separate the numeric range values). A greater-than/less-than search will test the specified data field of each individual record contained in the designated file for a numerical value greater-than or equal to the second and largest range value. Any records found within the specified range would be retrieved from the magnetic tape file and printed. For example, if a list of all currently enrolled students with a graduate QPR of 2.0 through 2.5 was wanted, the Current File would be searched using a data card containing L%20,25 as the search parameter key.

Any combination of permissible search parameter keys may be used with the following two restrictions:

1. Only two greater-than/less-than search keys may be used at one time. For example, if a list of all students born between 1956 and 1960, who received a masters degree, did not have a 1100 designator and received a degree in the area of management was wanted, the Historical File would be searched using data card(s) containing the following search parameter keys:

F%34,40
N%56,60
K=8
D#1100
J=M

The input data card image needed to conduct the above search could be as follows;

.....F%34,40.....N%56,60.....K=8.....D#1100.....J=M.....

The periods denote blank spaces.

2. A maximum of twenty search parameter keys may be used in any one file search. The search parameter keys are free-form as mentioned previously and can occur anywhere on the input data card(s) as long as one or more blank spaces separate each search key.

The error message, " ERROR IN INPUT CARDS * PROGRAM WILL NOT RUN * RECHECK INPUT CARDS AND RESUBMIT ", will occur if more than two greater-than/less-than search keys are used, more than a total of twenty search keys are used and when non-valid search key characters are used.

The possible combinations of search parameter keys are too numerous to list and it is felt by the authors that the record retrieval flexibility provided by SOIRS will enable the user to perform any search required.

Printed output contains all individual record data as listed in PART 2 for each record retrieved. Figures 12 through 17 are illustrations of sample output resulting from the following Historical and Current File searches:

1. FIGURE 12 is a partial output listing of a sample Current File search for designators not-equal to 9999, which will result in a complete printout of the file as there is no such designator. The input data card would contain: D#9999.

2. FIGURE 13 is output from a sample Current File when searched for USNR officers who do not have a P-Code of 1510. The input data card(s) would contain: C#1510.

3. FIGURE 14 is output from a sample Current File when searched for all 1100 designated officers with a graduate OPR of 2.0 through 3.0 from year groups 1955 through 1965. The input data card(s) would contain: D=1100 L%20,30 G%55,65.

4. FIGURE 15 is a partial output listing from a sample Current File when searched for Navy officers (USN and USNR) who have a graduate OPR of 1.7 through 3.0 and total OPR of 2.0 through 3.0. The input data card(s) would contain: E-N L%17,30 M%20,30.

5. FIGURE 16 is the complete output from the Historical File when searched for past graduates who received PhD's. The input data card would contain: K=9.

6. FIGURE 17 is a partial output listing from the Historical File when searched for 1100 designated past officer graduates who received a Masters degree, were enrolled in curriculum 360 and graduated in 1960 through 1965. The input data card(s) would contain: K=8 D=1100 N%60,65 J=360.

The curriculum officer, when initiating a file search, need only specify the parameters he desires the file to be searched for, and the type of search desired (equal, not-equal or greater-than/less-than); to the Application Programmer responsible for SOIRS. The Application Programmer will insure the input data cards contain the required search parameter keys and are keypunched correctly.

FIGURE 12 … FILE SEARCH PRINTED OUTPUT LISTING NUMBER 1

NAME RANK, FILE NO, SVC (CTY) (*) (*=F IF FEMALE, =1 IF IGEO)	SOCIAL SECURITY NUMBER	BIRTH DATE YR-M-D	DESIG (CHG)	RANK DATE YR-M-D	ORIG SOURCE CODE	YR GE	1ST RET YR	P CODE	NPGS QTR-YR ST/END	CUR RIC LUD	DEGR / AREA	OPR G/T	PAST DUTY STAS
JONES JOHN PAUL FRDM,767676,USN	007600751	280704	1100	620706	01-0	49	88		1-68 1-70	420	PHD OCEN	250 250	S PG MONTEREY USS CONSTITUTION USNA ANNAPOLIS USS CANVASSAIL NOC WASH DC USS NEVERSINK USS OARLOCK USS ROWBOAT
KILOWATT VERY READY ENS ,244813,USN	224466881	470406	1100	690601	01-0	68		8501	1-69 1-70	535	MS E.E.	242 263	S PG MONTEREY USNA ANNAPOLIS
LACY JOHNATHON PORSCHE LCDR,600002,USN	600092123	330302	1100	670501	01-0	60	79		3-67 2-69	360	MS AENG	219 228	S PG MONTEREY USS NUCKYPOO USS DIESLTYPE USNA ANNAPOLIS
LOCKER DAVY JONES CDR ,444111,USN	679367541	320606	1100	690204	01-0	64	73		3-68 3-70	426	MS MATS	205 239	S PG MONTEREY USS CANNYCAN CRUDESPAC USS TINCAN USS FAR CCUNTY MINEPAC MSO-35 MSC-53
LOCKSTEP PUNCHLY RICHARD LTJG,331155,USNR	331155770	420731	1105	691215	07-0	65			1-67 4-69	517	PHD MGT.	276 288	S PG MONTEREY
MASSA LAWRENCE LEE LCDR,625749,USN	545400532	340909	1600 (6201) (1100)	670101	21-1	58	73		3-67 2-69	360	MS AENG	239 214	NPGS MTY LST-1150 LST-1176 NPGS MTY DD-677
MCGILLACUDDY ELIOTEYES SALLY F LTJG,456123,USN	345678912	490505	1100	690505	07-0	67	87	1234	1-66 1-70	360	PHD FSCI	295 283	STA NUMBER ONE STA NUMBER TWO STA NUMBER THREE SYA NUMBER FOUR STA NUMBER FIVE STA NUMBER SIX
MOOSA LORENZO LEAN LCDR,628728,USN	628728123	340403	1100	660601	21-1	59	75		3-67 2-69	360	MS AENG	212 214	S PG MONTEREY USS FLATBOTTEM USS RUSTBUCKET USS FIRSTRIDE
PERUVIAN GUNFLASH DOMINIC COMD PERU		401212		630827					1-64 4-66	600	PHD C.E.	234 299	
PESAR JOHN ARNOLD LT ,828666,USN	828666666	361222	1405	680915	21-1	60	82	8362	4-67 1-70	410	PHD AENG	213 232	S PG MONTEREY USS CANBERRA SPECAST SECNAV NSYD SAN FRAN SERVLANT USS ORISKANY
PETERSON PERRY CANNONCOCKER LCOL USMC	424900030	310607		690314					3-68 4-70	527	MS C.E.	201 202	
PROTALONKA MICHELE RIMSOT LT ,824428,USNR	824428127	400430	1105	680521	07-0	65			3-67 2-69	360	MS AENG	280 285	S PG MONTEREY USS LONG BEACH
RESERVE ALWAYS READY LT ,842632,USNR	558877665	370224	1105	680226	07-0	60		8304	1-69 4-70	817	MS CMGT	210 215	S PG MONTEREY NRTC GLAKES ILL USS ROLLINGTUB NOB NORFOLK VA

CURRENT FILE SEARCH PRINTED OUTPUT LISTING NUMBER 2

NAME RANK, FILE NR, SVC/CTY(,*) (**F IF FEMALE,*1 IF IGEP)		SOCIAL SECURITY NUMBER	BIRTH DATE YR-M-D	DESIG (CHG)	RANK DATE YR-M-D	ORIG SOURCE CODE	YR GP	1ST RET YR	P CODE	NPGS QTR-YR ST/END	CUR RIC LUN	DEGR / AREA	QPR G/T	PAST DUTY STAS
CIVIL SERVICE DONOTHING CIV , ,USNR		111111111	420404			-				2-69 4-70	530	PG NONE	090 110	
COMMUNDER WAY DOWN GO ENS ,561234,USNR		234567891	490707	1300	650404	01-0	68			2-67 3-69	617	SYRC AENG	300 300	STA NUMBER ONE STA NUMBER TWO STA NUMBER THREE STA NUMBER FOUR STA NUMBER FIVE STA NUMBER SIX STA NUMBER SEVEN
LOCKSTEP FUNCHLY RICHARD LTJG,331155,USNR		331155779	420731	1105	681215	07-0	66			1-67 4-69	517	PV.. MG..	278 288	S PG MONTEREY
PROTALONKA MICHELE RUMSDT LT ,824428,USNR		824428123	400430	1105	690501	07-0	65			3-67 2-69	360	MS AENG	280 285	S PG MONTEREY USS LONG BEACH
RESERVE ALWAYS READY LT ,889432,USNR		998877665	370226	1105	650226	07-0	60		8304	1-69 3-70	817	MS CMGT	210 215	S PG MONTEREY NRTC GLAKES ILL USS [illegible] NOB NORFOLK VA
WIGGLEMORE SUZY SWIVELHIPS LTJG,891362,USNR	F	811362123	460406	1650	681203	19-0	70		9803	1-68 2-70	585	MS CHEM	225 232	S PG MONTEREY NAS CORONADO CAL
##########	END OF DATA		##########			END OF DATA			##########			END OF DATA		#########

FIGURE 13

CURR NR FIL. 3 ARCH PRINT D OUTPUT LISTING NUMBER 3

NAME RANK,FILE NR,SVC/CTY(,*) 1=F IE FEMALE,=I IE IGEP1	SOCIAL SECURITY NUMBER	BIRTH DATE YR=M=D	DESIG (CHGI	RANK DATE YR=M=D	ORIG SOURCE CODE	YR GP	LST RET YR	P CODE	NPGS QTR-YR STZEND	CUR RIC LUH	DEGR / AREA	OPR G/I	PAST DUTY STAS
ACADEMY JOE FRESHCAUGHT ENS ,608912,USN	109453867	481225	1100	680603	01-0	60	88		1-68 1-69	380	MS MATS	259 241	USNA S PG MONTEREY
AKERS MAXIMUM MOST LCDR,631892,USN	631892123	321206	1100	661225	01-0	58	79	8801	3-67 2-69	427	MS C.S.	278 276	S PG MONTEREY USS MOONFISH USS STARFISH USS TUNAFISH USS PINEFISH
HATCH WATER TIGHT CDR ,360497,USN	896735432	330505	1100 (6406) (1109)	690402	07-0	55	74		3-66 3-70	427	MS C.E.	230 205	S PG MONTEREY USS DIRTYDAN CRUDESLANT USS LEAKYCAN USS CLOSE COUNTY SERVEPAC MSO-26 MSC-62
HOARY ROBERT HEAVYSET LCDR,649231,USN	649231123	360603	1100 (6406) (1109)	681212	07-0	60	80		3-67 2-69	360	MS AENG	232 239	S PG MONTEREY USS SLOWMOVER 12NDHQ USS ROLYPOLY
LACY JOHNATHON PORSCHE LCDR,600992,USN	600992123	390309	1100	690601	01-0	60	79		3-67 2-69	360	MS AENG	219 228	S PG MONTEREY USS NUCKYPOO USS DIESLTYPE USNA ANNAPOLIS
MOOSA LORENZO LEAN LCDR,628728,USN	628728123	340403	1100	660601	21-1	58	75		3-67 2-69	360	MS AENG	212 214	S PG MONTEREY USS FLATBOTTEM USS RUSTBUCKET USS FIRSTRIDE
########## END OF DATA		########## END OF DATA			########## END OF DATA								##########

FIGURE 14

CURRENT FILE SEARCH PRINT: OUTPUT LISTING NUMBER 4

NAME RANK, FILE NR, SVC/CTY(,*) (*=F IF FEMALE,=I IF IGEP)	SOCIAL SECURITY NUMBER	BIRTH DATE YR-M-D	DESIG (CHG)	RANK DATE YR-M-D	ORIG SOURCE CODE	YR GP	1ST REV YR	P CODE	NPGS QTR-YR STZ/END	CUR RIC LUB	DEGR / AREA	OPR G/I	PAST DUTY STAS
ACADEMY JOE FRESHCAUGHT ENS ,608912,USN I	109453867	481225	1100	690603	01-0	60	88		1-68 1-69	380	MS MATS	259 241	USNA S PG MONTEREY
AKERS MAXIMUM MOST LCDR,631892,USN	631892123	321206	1100	651225	01-0	58	79	8801	3-67 2-69	427	MS C.S.	278 276	S PG MONTEREY USS MOONFISH USS STARFISH USS TUNAFISH USS PINEFISH
BADWIN RONALDO SQUARE LT ,269052,USN	269052123	390309	1400 (6901) (1105)	590226	07-0	64			3-67 2-69	360	MS AENG	285 290	S PG MONTEREY MACV USS AMERICA
BARNALD ROBERTO SKINHEAD LT ,723614,USN	723614123	350803	1320	671215	07-0	62	91		3-67 2-69	360	MS AENG	225 232	S PG MONTEREY NAS WHIDBY NAS ALAMEDA
DOWNUNDER WAY DOWN GO ENS ,541234,USNR	234567891	490707	1300	680404	01-0	68			2-67 3-69	817	3YRC AENG	300 300	STA NUMBER ONE STA NUMBER TWO STA NUMBER THREE STA NUMBER FOUR STA NUMBER FIVE STA NUMBER SIX STA NUMBER SEVEN
HATCH WATER TIGHT CDR ,360497,USN	896735432	330505	1100 (6406) (1105)	690402	07-0	55	74		3-68 3-70	427	MS C.E.	239 205	S PG MONTEREY USS DIRTYPAN CRUDESLANT USS LEAKYCAN USS CLOSE COUNTY SERVEPAC MSO-26 MSC-62
HENRY ROBERT LEE LCDR,631894,USN	449426211	360717	1400 (6901) (1100)	671010	07-0	59	79		1-67 2-69	360	MS AENG	240 222	NPGS MTY DD-728 CNSF SFRAN DER-389
HOARY ROBERT HEAVYSET LCDR,649231,USN	649231123	360603	1100 (6406) (1105)	581212	07-0	60	80		3-67 2-69	360	MS AENG	232 239	S PG MONTEREY USS SLOWMOVER 12NDHQ USS ROLYPOLY
JONES JOHN PAUL FADM, 67676,USN	007600761	280706	1100	620706	01-0	49	88		1-68 1-70	420	PHD OCFN	250 250	S PG MONTEREY USS CONSTITUTION USNA ANNAPOLIS USS CANVASSAIL NOC WASH DC USS NEVERSINK USS OARLOCK USS ROWBOAT
KILOWATT VERY READY ENS ,244813,USN I	224466881	470606	1100	690601	01-0	68		8801	1-69 1-70	535	MS F.E.	242 263	S PG MONTEREY USNA ANNAPOLIS
LACY JOHNATHON PORSCHE LCDR,600992,USN	600992123	390309	1130	690601	01-0	60	79		3-67 2-69	360	MS AENG	219 228	S PG MONTEREY USS NUCKYPOO USS DIESLTYPE USNA ANNAPOLIS
LOCKER DAVY JONES CDR ,444111,USN	679367541	320606	1100	680206	01-0	54	73		3-68 3-70	426	MS MATS	205 239	S PG MONTEREY USS CANNYCAN CRUDESPAC USS TINCAN USS FAR COUNTY MINEPAC

FIGURE 15

HISTORICAL FILE SEARCH PRINTED OUTPUT LISTING NUMBER 1

NAME RANK,FILE NO,SVC/CTY(,*) (*=F IF FEMALE,=I IF IGEP)	SOCIAL SECURITY NUMBER	BIRTH DATE YR-M-D	DESIG (CHG)	RANK DATE YR-M-D	ORIG SOURCE CODE	YR GP	1ST RET YR	P CODE	NPGS QTR-YR ST/END	CUR RYC LUM	DEGR / AREA	OPR G/I	PAST DUTY STAS
ANDERSON JAMES BRUCE LCDR,548434,USN	568353625	331016	1510		04-0	56		8106	-66	610	PHD A.E.		MISLCEN PT MUGU
~~CTH ROGER GRAYDON CDR ,569851,USN	381267495	290729	1310		01-0	52		8108	-67	530	PHD PHY		VP 9
BUSH THOMAS OLIVER LT ,638827,USN	441369461	380714	1700 (6807) (110C)		04-0	60			-68	521	PHD PHY		DEF ATOM SUPPAGY
FEATHERSTON FRANK HUNTER CDR ,532795,USN	224523892	290309	1510		01-0	50			-63	530	PHD PHY		NASC AIR 01
GALL CAUNE MIGUEL CDR ,491369,USN	5C4183312	290501	1510 (6456) (1100)		06-0	53			-65	530	PHD PHY		NASC AIR 03
HARRIS JACK REDVER CDR ,514936,USN	367320053	301017	1510		04-0	53		8205	-65	590	PHD E.E.		NADC NAF JHNSVL
HANT CAVID AUSTIN LCDR,630564,USN	425686942	380224	1510 (6700) (1350)		06-0	60			-66	380	PHD PHY		RVAW 120
HOFFMANN JOHN MELVILLE LCDR,620079,USN	178264430	341205	1510 (6506) (1310)		03-0	58			-66	530	PHD CHEM		NAVWEAPCEN CHILK
HOLLISTER FLOYD HILL LCDR,580122,USN	132282225	370428	1400 (6306) (1100)		04-0	58			-65	590	PHD E.E.		NELECLABCEN SDGO
HUDSON RALPH EDWARD LT ,639877,USN	559500334	361121	1510 (6506) (1515)		06-0	60			-66	530	PHD E.E.		VO 1
JAUREGUI STEPHEN JR CDR ,552548,USN	562209287	25C630	1610 (5208) (640C)		22-1	54			-62	590	PHD E.E.		NSA FT MEADE MD
KIMBLE CHARLES DONALD LCDR,592322,USN	523467922	371012	1310 (6206) (1100)		04-0	59		8301	-68	530	PH- CHE-		CVA 34 ORISKANY
LEVIN RICHARD ROBERT LCDR,558991,USN	073285793	341110	1510 (6604) (1310)		04-0	57			-66	530	PHD PHY		NASPASYSACT LOSA
MILLER ALOYSIUS RUDOLPH LCDR,579915,USN	146282459	360526	1400 (6306) (1100)		04-0	58			-65	570	PHD E.E.		NSYD CHARLESTN
PHELPS PHARO ALFRED CDR ,533099,USN	561305700	280824	5100		01-0	50			-63	380	PHD PHY		OICC NFECC SPAIN
POPE JOHN WILLIAM RIPPON JR CDR ,569604,USN	410642126	310515	1610 (6106) (1615)		07-0	54			-66	590	PHD E.E.		NAV SGRCOMDHC DC
STURM EDWARD JOHN LT ,593085,USN	296303136	370610	1510 (6700) (1320)		04-0	60		8106	-68	610	PHD A.E.		NASC AIR 051
TUCKER JAMES EARL LCDR,561119,USN	514206269	320609	1510 (6510) (1510)		04-0	56			-66	530	PHD PHY		MISLCEN PT MUGU

FIGURE 16

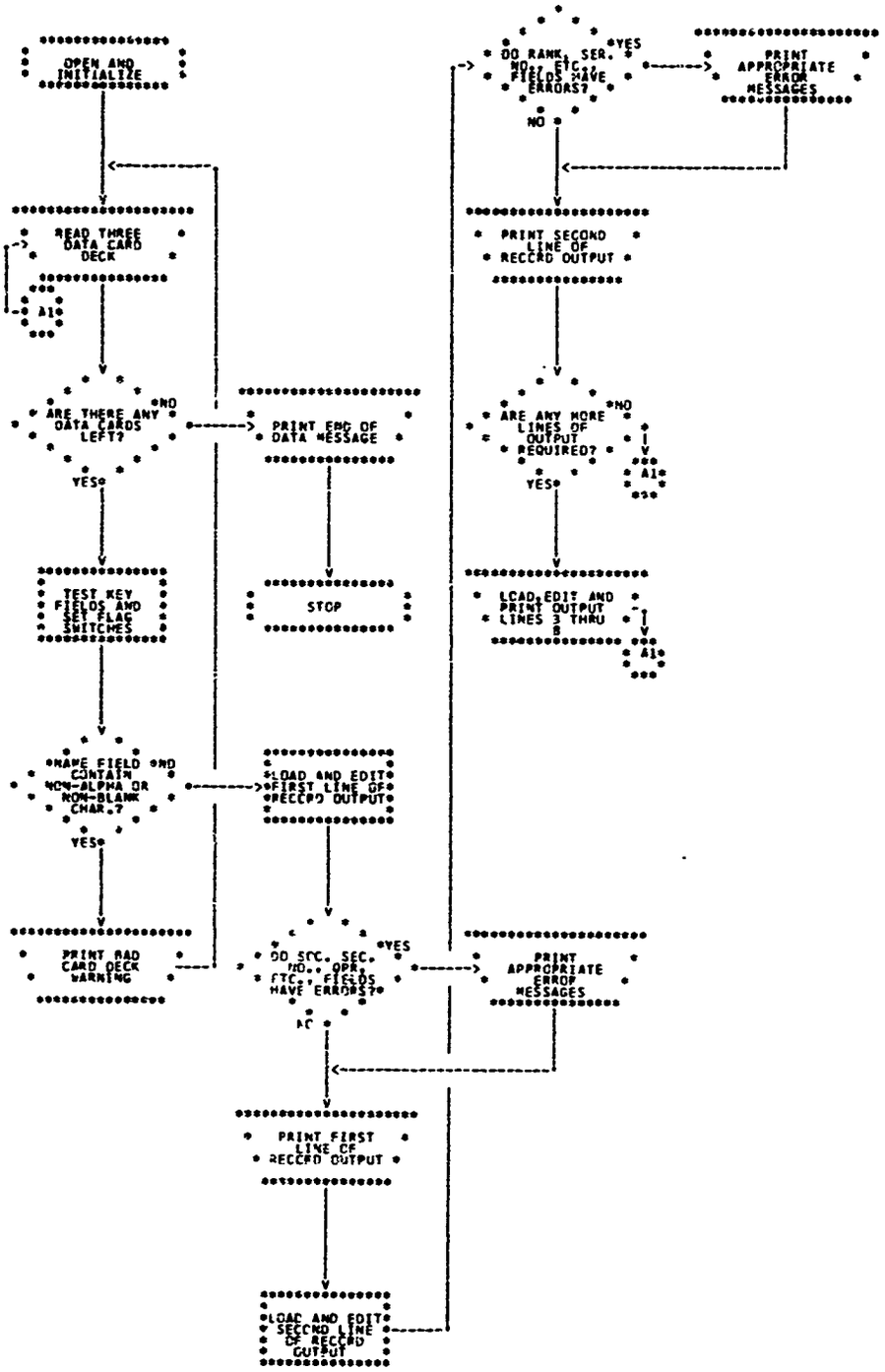
BIOGRAPHICAL FILE SEARCH PRINTED OUTPUT LISTING NUMBER 2

NAME RANK,FI NR,SVC/CITY(*) (0=F IF FEMALE,=I IF IGEP)	SOCIAL SECURITY NUMBER	BIRTH DATE YR-M-D	DESIG (CHG)	RANK DATE YR-M-D	ORIG SOURCE CODE	YR GP	1ST RET YR	P CODE	NPGS QTR-YR SY/END	CUR PIC LUM	DEGR / AREA	OPR GZT	PAST DUTY STAS
AILES ROBERT HERER LCDR,613310,USN	224525011	351024	1100		01-0	57		8501	-83	360	MS O.R.		DE1015 HAMERBERG
BALDWIN EDWIN MCLEAN LCDR,424325,USN	397327398	361019	1100		01-0	58		8501	-84	360	MS O.R.		NPTU IDAHO FALLS
BEATTY JAMES ROBERT III LCDR,613369,USN	179281450	350711	1100		01-0	57		8501	-85	360	MS O.R.		SS 567 GUDGEON
BERNSTEIN PAUL ALBERT LCDR,464723,USN	015265182	350726	1100		04-0	57		8501	-84	360	MS O.R.		FLTCOMPROGCENLNT
BRANCH ALVIN DEON CDR ,542457,USN	224521335	280624	1100		01-0	50		8501	-83	360	MS O.R.		DDG 6 BARNEY
COLLINS FERDINAND IGLEHART JR CDR ,584391,USN	136326601	320226	1100		01-0	54		8501	-81	360	MS O.R.		OPNAV
CONWAY JAMES MCNARNEY LCDR,594988,USN	364264704	320201	1100		01-0	55		8501	-83	360	MS O.R.		SS 350 DOGFISH
CRANE JOHN MILLOUGHBY JR CAPT,484053,USN	420228070	250613	1100		01-0	46		8501	-80	360	MS O.R.		COM PHIB GR 2
CRAWFORD WILLIAM TRAVIS CDR ,584402,USN	410642013	310607	1100		01-0	54		8501	-81	360	MS O.R.		DE1049 KOELSCH
CUMMINS PAUL ZACH II LCDR,624445,USN	074288719	360501	1100		01-0	58	69	8501	-84	360	MS O.R.		OPNAV
DAVIS CHARLES EDWARD LCDR,634227,USN	108305058	371103	1100 (6406) (1105)		02-0	60		8501	-85	360	MS O.R.		NAVAL ACAD
GARDNER GEOFFREY LELAND LCDR,594938,USN	375407074	330420	1100		01-1	55		8501	-83	360	MS O.R.		BUPERS PERS N
GREENWALSH WILLIAM TAYLOR JR LCDR,594953,USN	228424011	340121	1100		01-0	55		8501	-84	360	MS O.R.		OSD
HAGER CHARLES FRANKLIN CDR ,586700,USN	184240050	321218	1100		06-0	54		8501	-84	360	MS O.R.		NOSC OPD 06
HALL JOHN VALENTINE CDR ,528593,USN	271544472	321224	1100		04-0	54		8501	-83	360	MS O.R.		OPNAV
HODGSKISS WILLIAM LAWRENCE LCDR,545236,USN	516347647	330628	1100		04-0	56		8501	-85	360	MS O.R.		COMSERVPAC
HOWE JOHN EDWARD LCDR,601486,USN	055482116	350619	1100 (6206) (1105)		06-0	57		8501	-85	360	MS O.R.		COM MSTS
HUGHES WAYNE PHILIP JR CDR ,554984,USN	339246805	300530	1100		01-0	52		8501	-84	360	MS O.R.		DD 948 HORTON
HUGHES THOMAS JOSEPH JR CAPT,472573,USN	068138099	261014	1100		21-1	46		8501	-82	360	MS O.R.		AO 54 CHIKASKIA
JOHNSTON JOHN MACKINTOSH LCDR,570061,USN	482305270	311207	1100		01-0	53		8501	-84	360	MS O.R.		MAR COL NEWPORT
KEANE JAMES PHILIP CDR ,595003,USN	018222502	290114	1100		01-0	52		8501	-80	360	MS O.R.		SSBN 634 BLUE CR
KRENZER JOSEPH JOHN JR LCDR,598017,USN	207242216	330501	1100		01-0	55		8501	-83	360	MS O.P.		SS 365 HAWKBILL

FIGURE 17

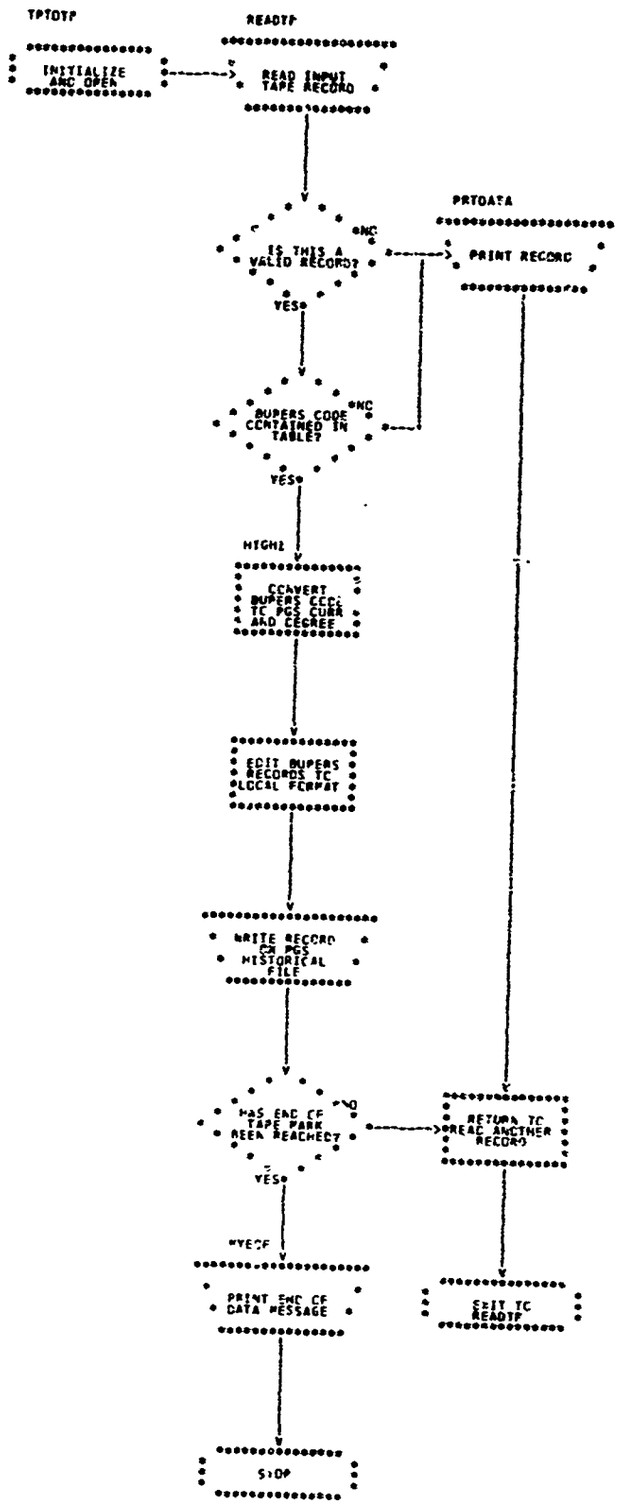
*** APPENDIX B -- FLOWCHART NUMBER 1 ***

DATA CARD DECK VERIFICATION PROGRAM



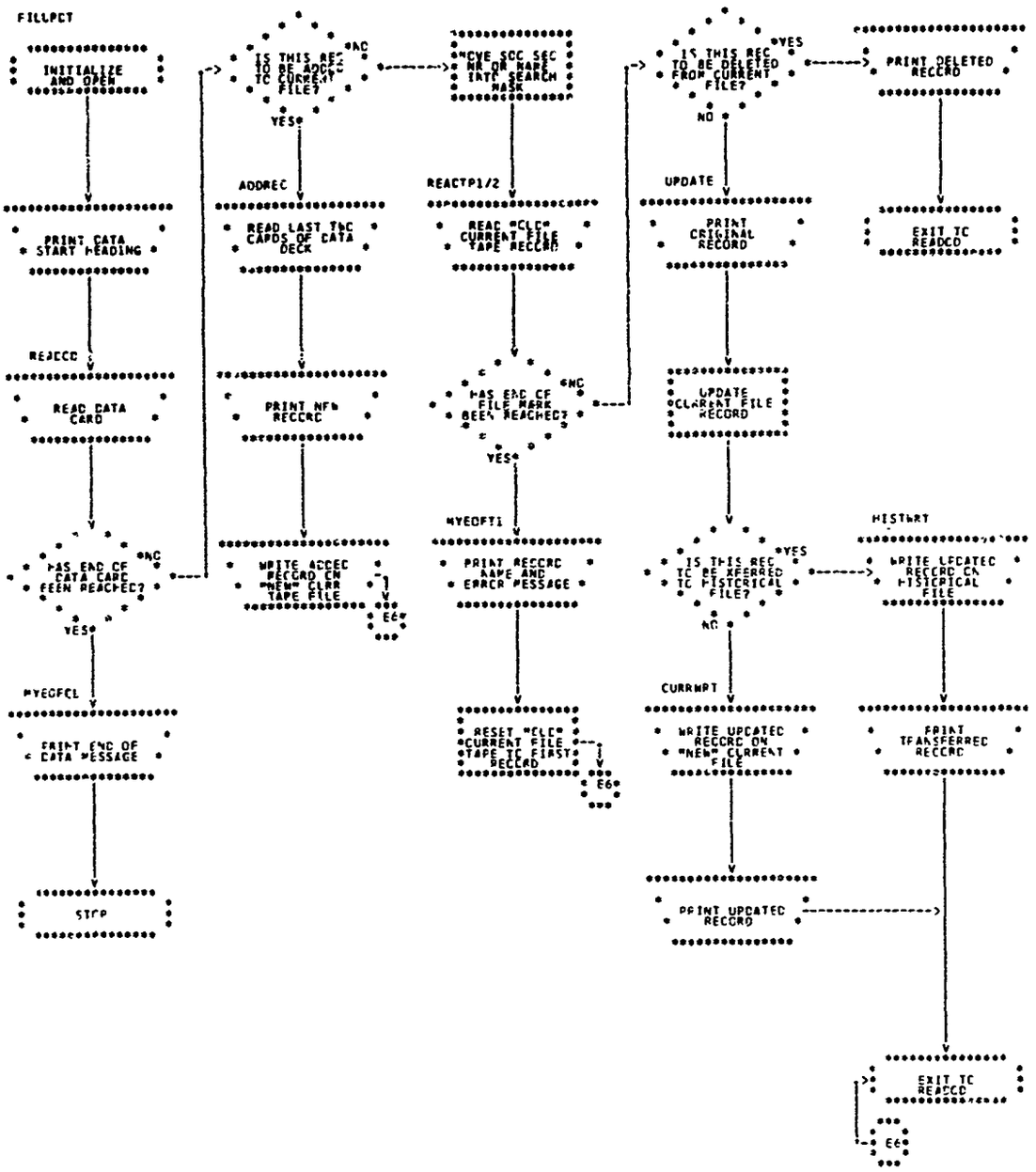
*** APPENDIX B -- FLOWCHART NUMBER 2 ***

BUPERS TAPE DATA TRANSFER PROGRAM



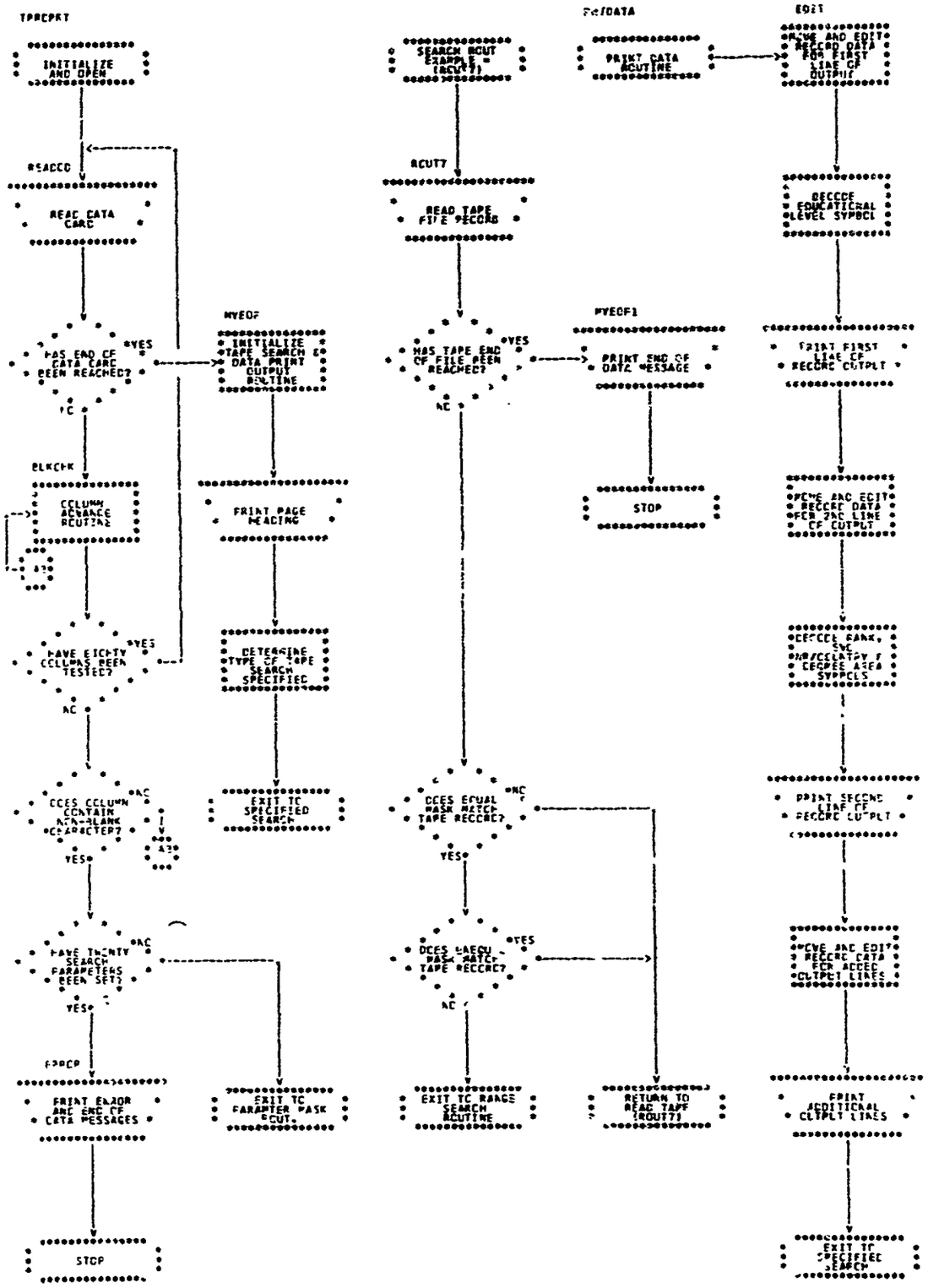
*** APPENDIX B -- FLOWCHART NUMBER 3 ***

CURRENT TAPE FILE UPDATE AND TRANSFER PROGRAM



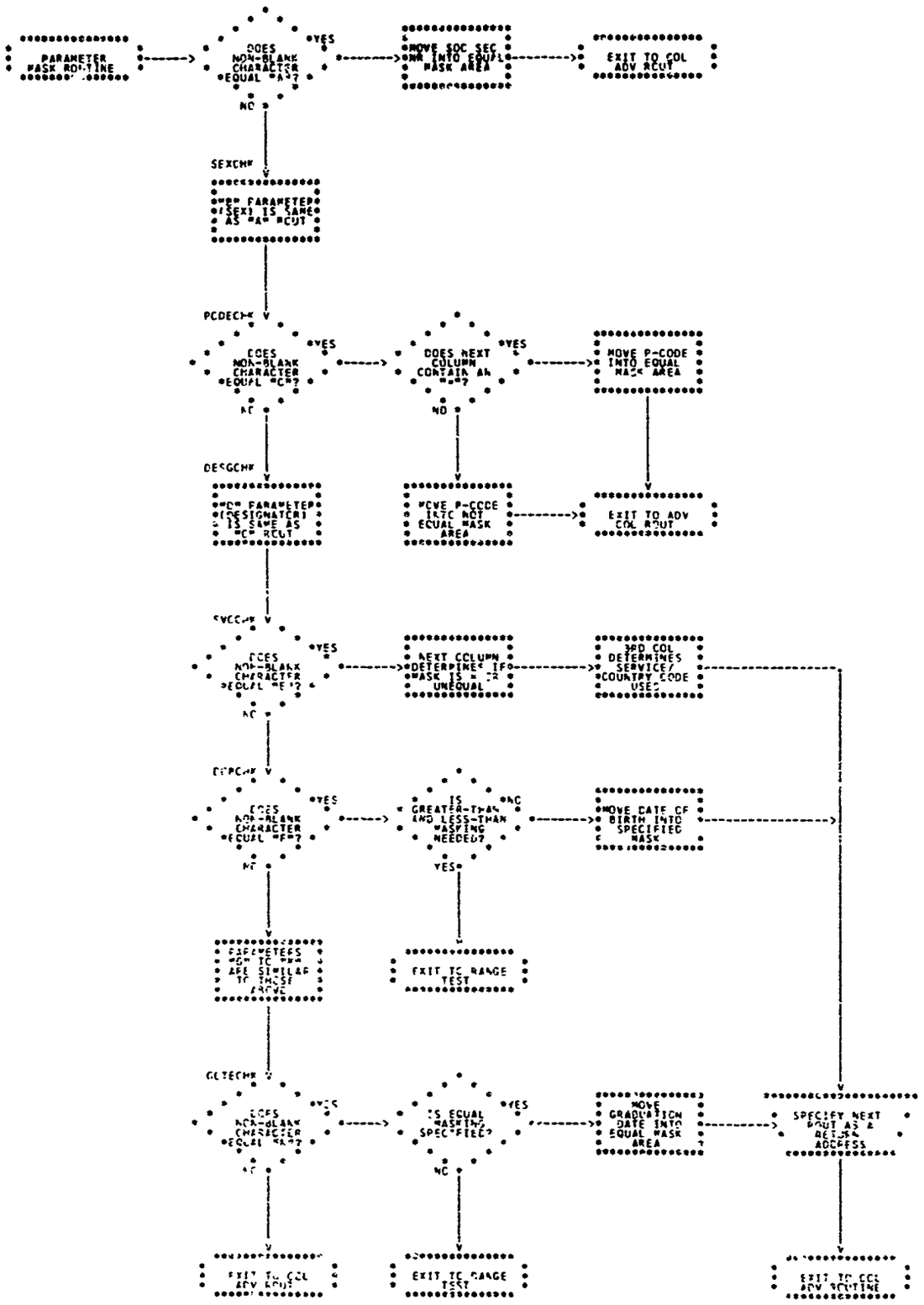
*** APPENDIX B -- FLOWCHART NUMBER 4A ***

DATA FILE SEARCH PROGRAM



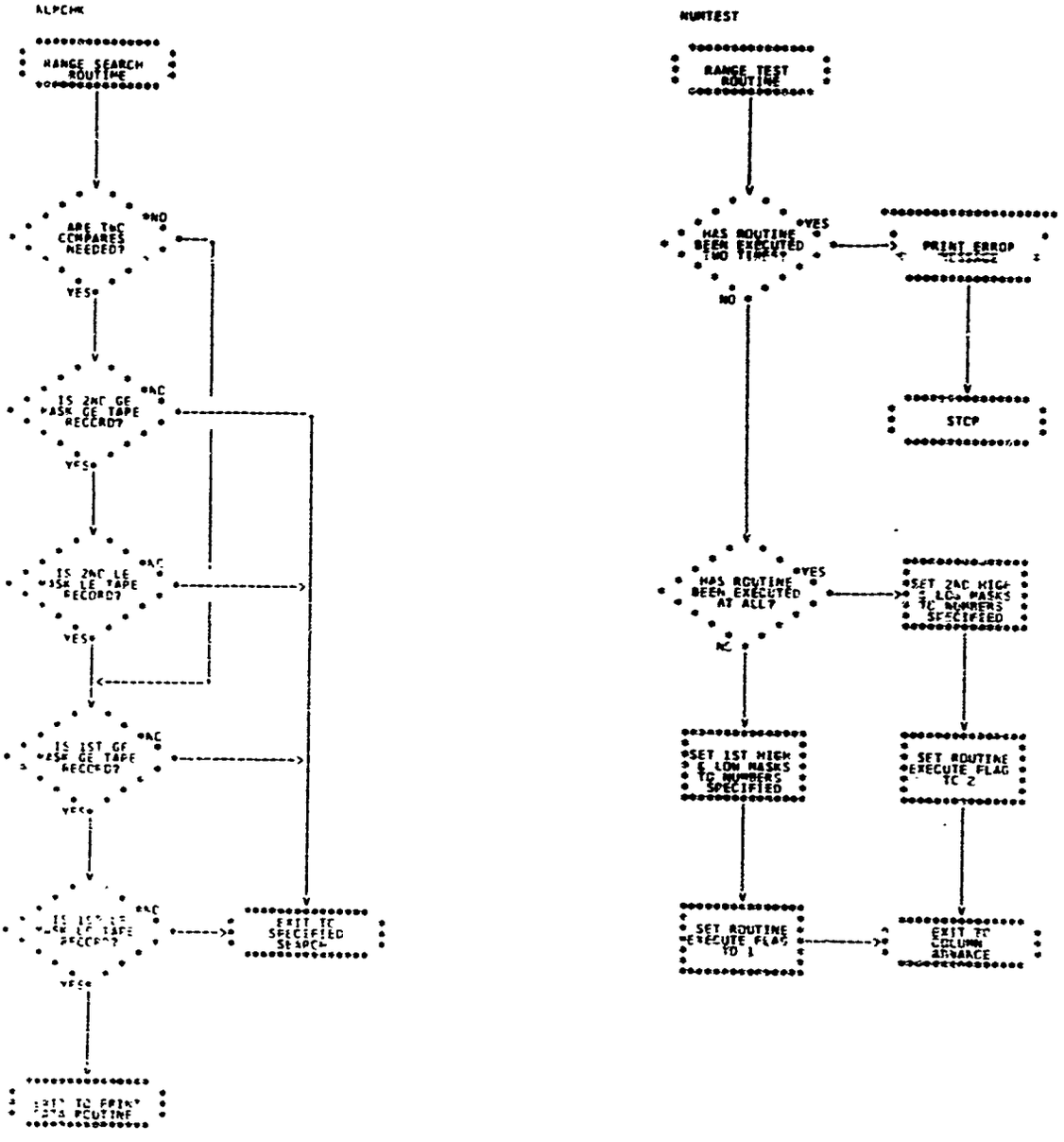
*** APPENDIX B -- FLOWCHART NUMBER 48 ***

DATA FILE SEARCH PROGRAM



*** APPENDIX B — FLOWCHART NUMBER 4C ***

DATA FILE SEARCH PROGRAM



```
// EXEC ASMALG
//ASM.SYSIN DD *                    APPENDIX B   CRDCHK1
*********************************************************************
*********************************************************************
*********************************************************************
**                                                                 **
**                                                                 **
** THIS PROGRAM WILL READ USNPGS STUDENT OFFICER INFORMATION       **
** RETRIEVAL SYSTEM DATA CARD DECKS CONSISTING OF THREE CARDS PER  **
** INDIVIDUAL RECORD. DATA WILL BE DECODED, EDITED AND PRINTED     **
** IN NORMAL OUTPUT FORMAT WITH ERROR MESSAGES PRINTED FOR KEY BAD **
** DATA. ALL ERRORS MUST BE CORRECTED PRIOR TO STORING DATA ON TAPE. **
**                                                                 **
**                                                                 **
*********************************************************************
*********************************************************************
*********************************************************************

*          ***************************************************
*          ***************************************************
*          **********                             **********
*          **********  INITIALIZATION AND OPENING ROUTINE **********
*          **********                             **********
*          ***************************************************
*          ***************************************************

CRDCHK1  CSECT
         USING *,15
         SAVE  (14,12),T,*
         LR    12,13
         CNOP  0,4
         BAL   13,*+76
         DROP  15
         USING *,13
         DS    18F
         ST    13,8(12)
         ST    12,4(13)
         OPEN  (IN,(INPUT),OUT,(OUTPUT))
         LA    5,LINE                    R5 SET TO LINE OUTPUT ADDRESS
         LA    7,CRDAREA                 R7 SET TO CARD 1 INPUT ADDRESS
         LA    8,80(7)                   R8 SET TO CARD 2 INPUT ADDRESS
         LA    9,160(7)                  R9 SET TO CARD 3 INPUT ADDRESS
         LA    10,1                      R10 SET TO CONSTANT VALUE 1
         LA    11,2                      R11 SET TO CONSTANT VALUE 2
         B     *+8                       BRANCH AROUND PREVIOUS PAGE FLAG
TITLE    MVI   TEST,C'0'                 SET PREVIOUS PAGE FLAG
         LA    6,70                      SET LINE CTR (R6) TO 70 LINES
```

```
         PUT   OUT,TITL1              PRINT 1ST LINE OF PAGE HEADING
         PUT   OUT,TITL2              PRINT 2ND LINE OF PAGE HEADING
         PUT   OUT,TITL3              PRINT 3RD LINE OF PAGE HEADING
         PUT   OUT,TITL4              PRINT PAGE HEADING SEPARATER
         CLI   TEST,C'$'              TEST PREVIOUS PAGE FLAG
         BC    8,EDIT                 COND BRANCH AROUND CD READ AREA
*        ******************************************************************
*        **********                                              **********
*        **********               READ DATA DECK                 **********
*        **********                                              **********
*        ******************************************************************
READ     GET   IN,(7)                 READ 1ST CARD
         LR    7,1                    STORE CARD 1 ADDR IN R7
         GET   IN,(8)                 READ 2ND CARD
         LR    8,1                    STORE CARD 2 ADDR IN R8
         GET   IN,(9)                 READ 3RD CARD
         LR    9,1                    STORE CARD 3 ADDR IN R9
*        ******************************************************************
*        **********                                              **********
*        ********** TEST KEY FIELDS AND SET FLAG SWITCHES        **********
*        **********                                              **********
*        ******************************************************************
         LR    2,7                    STORE CARD 1 ADDR IN R2
         LA    3,10                   SET R3 TO CONSTANT VALUE TEN
         AR    2,10                   ADD 1 TO CARD 1 ADDR
         CLI   5(2),X'40'             TEST NAME FIELD FOR BLANK CHAR
         BC    8,*+20                 COND BRANCH IF BLANK
         CLI   5(2),C'A'              TEST NAME FIELD FOR SPECIAL CHAR
         BC    4,*+20                 COND BRANCH TO BAD CARD ROUT
         CLI   5(2),C'Z'              TEST NAME FIELD FOR NUMERIC CHAR
         BC    2,*+12                 COND BRANCH TO BAD CARD ROUTINE
         BCT   3,*-24                 BRANCH FOR NEXT CHAR TEST
         B     *+24                   BRANCH AROUND BAD CARD DECK ROUT
         SP    6,11                   SUBTRACT 1 FROM LINE CTR
         PUT   OUT,WARNING            PRINT BAD CARD DECK WARNING
         B     READ                   BRANCH TO READ NEXT DECK
         MVC   TEST,23(8)             MOVE DUTY STA NR INTO TEST AREA
         NI    TEST,X'0F'             CONVERT DUTY STA CODE TO BINARY
         SR    12,12                  CLEAR R12
         IC    12,TEST                INSERT BINARY NR IN R12
         CLI   23(8),C'8'             TEST MAX VALUE OF DUTY STA NR
         BC    2,*+12                 COND BRANCH TO ERROR MSG ROUT
         CLI   23(8),C'0'             TEST MIN VALUE OF DUTY STA NR
         BC    8,ERR*-6               COND BRANCH IF FOREIGN STUDENT
```

```
         BC    11,TEST1                  COND BRANCH AROUND ERR MSG ROUT
         MVC   ELINE+27(1),23(8)         MOVE INCORRECT NR INTO ERROR MSG
         MVC   ELINE+16(4),=C'DUTY'      INSERT DUTY INTO ERROR MSG
         MVI   ELINE,C'0'                DOUBLE SPACE ERROR LINE
         SR    6,11                      SUBTRACT 2 FROM LINE CTR
         PUT   OUT,ELINE                 PRINT DUTY STA ERROR MSG
         LA    12,3                      SET DUTY STA NR TO 3
TEST1    CLI   0(8),C'A'                 TEST 4TH DUSTA FOR MIN VALID LTR
         BC    4,*+12                    COND BRANCH TO 3 DUTY STAS
         CLI   0(9),C'7'                 TEST 4TH DUSTA FOR MAX VALID LTR
         BC    13,*+12                   COND BRANCH AROUND 3 DUTY STAS
         LA    12,3                      SET DUTY STA NR TO 3
         MVI   DBCK,C'*'                 SET 2 CARD DECK FLAG
         CR    12,6                      TEST LINES REMAINING ON PAGE
         BC    2,TITLE                   COND BRANCH TO NEW PAGE
         B     EDIT                      BRANCH AROUND FOREIGN STDNT ROUT
         CR    11,6                      TEST FOR 2 LINES LEFT ON PAGE
         BC    2,TITLE                   COND BRANCH TO NEW PAGE

*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********  'LOAD EDIT AND PRINT'ROUTINE FOR LINE         **********
*        **********              ONE DATA OUTPUT                   **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************

EDIT     MVI   0(5),C' '                 INSERT BLANK CHAR IN OUTPUT LINE
         MVC   1(132,5),0(5)             CLEAR OUTPUT LINE
         CLI   ELINE,C'0'                TEST FOR PRIOR ERROR MSG
         BC    8,*+8                     COND BRCH AROUND DOUBLE SPACING
         MVI   0(5),C'0'                 INSERT DOUBLE SPACE CHARACTER
         MVC   1(31,5),6(7)              MOVE NAME INTO OUTPUT LINE
         MVC   45(6,5),58(7)             MOVE DOR INTO OUTPUT LINE
         MVC   61(6,5),48(7)             MOVE DOR INTO OUTPUT LINE
         CLI   21(8),C'1'                TEST FOR NON-U.S.OFFICER
         BC    2,FOR*TEST                COND BRANCH AROUND U S ENTRIES

*        ********************************************************************
*        **********                                                **********
*        **********        SOCIAL SECURITY NUMBER CHECK            **********
*        **********                                                **********
*        ********************************************************************

         LR    2,7                       STORE CARD 1 ADDR IN REG 2
         LA    3,9                       INSERT CONSTANT 9 IN REG 4
```

```
         AP    2,10                    ADD 1 TO CARD 1 ADDR
         CLI   64(2),C'0'              TEST SSNR FIELD FOR VALID CHAR
         BC    4,*+12                  COND BRANCH TO SSNR ERROR ROUT
         BCT   3,*-10                  BRANCH AND TEST NEXT CHARACTER
         B     *+60                    BRANCH AROUND ERROR ROUTINE
         CLI   ELINE,C'0'              TEST FOR PRIOR ERROR MESSAGE
         BC    8,*+12                  COND BRANCH TO SINGLE SPACING
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         B     *+8                     BRANCH AROUND SINGLE SPACING
         MVI   ELINE,C' '              SINGLE SPACE ERROR LINE
         MVC   ELINE+27(1),64(2)       MOVE INCORRECT CHAR INTO ERR MSG
         MVC   ELINE+16(4),=C'SSNR'    INSERT 'SSNR' INTO ERROR MESSAGE
         SP    6,10                    SUBTRACT ONE FROM PAGE LINE CTR
         PUT   OUT,ELINE               PRINT SOC SEC NR ERROR MESSAGE
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         MVI   0(5),C' '               SINGLE SPACE OUTPUT LINE
         MVC   33(9,5),65(7)           MOVE SSNR INTO OUTPUT LINE
         CLI   23(8),C'0'              TEST FOR FOREIGN OFFICER
         BC    8,FORTEST               COND BRANCH AROUND US ONLY ITEMS

*        ********************************************************************
*        **********                                              **********
*        **********              DESIGNATOR CHECK                **********
*        **********                                              **********
*        ********************************************************************

         LR    2,7                     STORECARD 1 ADDR IN R2
         LA    3,4                     INSERT CONSTANT 4 IN R4
         AR    2,10                    ADD 1 TO CARD 1 ADDR
         CLI   37(2),C'0'              TEST DESIG FIELD FOR VALID CHAR
         BC    4,*+12                  COND BRANCH TO DESIG ERR ROUT
         BCT   3,*-10                  BRANCH FOR NEXT CHAR TEST
         B     *+60                    BRANCH AROUND ERROR ROUTINE
         CLI   ELINE,C'0'              TEST FOR PRIOR ERROR MESSAGE
         BC    8,*+12                  COND BRANCH TO SINGLE SPACING
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         B     *+8                     BRANCH AROUND SINGLE SPACING
         MVI   ELINE,C' '              SINGLE SPACE ERROR LINE
         MVC   ELINE+27(1),37(2)       MOVE INCORRECT CHAR INTO ERR MSG
         MVC   ELINE+16(4),=C'DESG'    INSERT DESG INTO ERROR MSG
         SP    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE               PRINT DESG ERROR MSG
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         MVI   0(5),C' '               SINGLE SPACE OUTPUT LINE
         MVC   54(4,5),38(7)           MOVE DESIGNATOR INTO OUTPUT LINE
         CLI   45(7),C' '              TEST FOR SOURCE CODE ENTRY
         BC    8,*+20                  COND BRANCH AROUND SRCE CDE ROUT
         MVC   70(2,5),45(7)           MOVE SOURCE CODE INTO OUTPT LINE
```

```
         MVI   72(5),C'-'                 INSERT SPACER
         MVC   73(1,5),47(7)              MOVE SRCE CODE MOD TO OUTPT LINE
         MVC   77(2,5),42(7)              MOVE YEAR GROUP INTO OUTPUT LINE
         MVC   82(2,5),74(7)              MOVE 1ST RET YR INTO OUTPUT LINE
         MVC   87(4,5),54(7)              MOVE P-CODE INTO OUTPUT LINE

*        ********************************************************************
*        **********                                              **********
*        **********          SCHOOL START DATE CHECK             **********
*        **********                                              **********
*        ********************************************************************

FORTEST  LR    2,8                        STORE CARD 2 ADDR IN R2
         LA    3,3                        INSERT CONSTANT 3 IN R3
         AR    2,10                       ADD 1 TO CARD 2 ADDR
         CLI   7(2),C'0'                  TEST START DATE FOR VALID CHAR
         BC    4,*+12                     COND BRANCH TO ERROR ROUT
         BCT   3,*-10                     BRANCH FOR NEXT CHAR TEST
         B     **+60                      BRANCH AROUND ERROR ROUTINE
         CLI   ELINE,C'0'                 TEST FOR PRIOR ERROR MESSAGE
         BC    8,*+12                     COND BRANCH TO SINGLE SPACING
         MVI   ELINE,C'0'                 DOUBLE SPACE ERROR LINE
         B     **+8                       BRANCH AROUND SINGLE SPACING
         MVI   ELINE,C' '                 SINGLE SPACE ERROR LINE
         MVC   ELINE+27(1),7(2)           MOVE INCORRECT CHAR TO ERR MSG
         MVC   ELINE+16(4),=C'SDTE'       INSERT SDTE   TO ERR MSG
         SP    6,10                       SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE                  PRINT START DATE ERR MSG
         MVI   ELINE,C'0'                 DOUBLE SPACE ERROR LINE
         MVI   0(5),C' '                  SINGLE SPACE OUTPUT LINE
         MVC   94(1,5),8(8)               MOVE START QTR INTO OUTPUT LINE
         MVI   95(5),C'-'                 INSERT SPACER
         MVC   96(2,5),9(8)               MOVE START YR INTO OUTPUT LINE

*        ********************************************************************
*        **********                                              **********
*        **********              CURRICULUM CHECK                **********
*        **********                                              **********
*        ********************************************************************

         LR    2,7                        STORE CARD 1 ADDR IN R2
         LA    3,3                        INSERT CONSTANT 3 IN R3
         AR    2,10                       ADD 1 TO CARD 1 ADDR
         CLI   75(2),C'0'                 TEST CURRICULUM FOR VALID CHAR
         BC    4,*+12                     COND BRANCH TO CURR ERR ROUT
         BCT   3,*-10                     BRANCH FOR NEXT CHAR TEST
         B     **+60                      BRANCH AROUND ERROR ROUTINE
         CLI   ELINE,C'0'                 TEST FOR PRIOR ERROR MESSAGE
```

```
         BC    8,*+12                  COND BRANCH TO SINGLE SPACING
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         B     *+8                     BRANCH AROUND SINGLE SPACING
         MVI   ELINE,C' '              SINGLE SPACE ERROR LINE
         MVC   ELINE+27(1),75(2)       MOVE INCORRECT CHAR INTO ERR MSG
         MVC   ELINE+16(4),=C'CURR'    INSERT CURR INTO ERR MSG
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE               PRINT CURRICULUM ERROR MSG
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         MVI   0(5),C' '               SINGLE SPACE OUTPUT LINE
         MVC   101(3,5),76(7)          MOVE CURRICULUM INTO OUTPUT LINE

*        ***************************************************************
*        **********                                           **********
*        **********          EDUCATIONAL LEVEL CHECK          **********
*        **********                                           **********
*        ***************************************************************

         CLI   79(7),X'F0'             TEST DEGREE CODE FOR VALID NR
         BC    11,*+64                 COND BRANCH AROUND DEGR ERR ROUT
         CLI   ELINE,C'0'              TEST FOR PRIOR ERROR MESSAGE
         BC    8,*+12                  COND BRANCH TO SINGLE SPACING
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         B     *+8                     BRANCH AROUND SINGLE SPACING
         MVI   ELINE,C' '              SINGLE SPACE ERROR LINE
         MVC   ELINE+27(1),37(7)       MOVE BAD DEGR CODE NR IN ERR MSG
         MVC   ELINE+16(4),=C'DEGR'    INSERT DEGR INTO ERROR MSG
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE               PRINT DEGR ERROR MSG
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         MVI   0(5),C' '               SINGLE SPACE OUTPUT LINE
         B     *+30                    BRANCH AROUND DEGREE DECODING
         NI    79(7),X'0F'             CONVERT DEGR CODE TO BINARY NR
         SR    2,2                     CLEAR R2
         IC    2,79(7)                 INSERT DEGR CODE IN R2
         SLL   2,2                     MULTIPLY DEGR CODE BY 4
         LA    4,TABLE1                PUT DEGR TABLE ADDR IN R4
         AR    2,4                     SET R2 TO CORRECT DEGR ADDR
         MVC   106(4,5),0(2)           MOVE DEGR INTO OUTPUT LINE

*        ***************************************************************
*        **********                                           **********
*        **********          GRADUATE 'OPP' CHECK             **********
*        **********                                           **********
*        ***************************************************************

         LR    2,8                     STORE CARD 2 ADDR IN R1
         LA    3,3                     INSERT CONSTANT 3 IN R3
```

```
         AP    2,10                    ADD 1 TO CARD 2 ADDR
         CLI   13(2),C'0'              TEST GRAD OPR FOR VALID CHAR
         BC    4,*+12                  COND BRANCH TO ERROR ROUT
         BCT   3,*-10                  BRANCH FOR NEXT CHAR TEST
         B     **+60                   BRANCH AROUND ERROR ROUTINE
         CLI   ELINE,C'0'              TEST FOR PRIOR ERROR MESSAGE
         BC    8,*+12                  COND BRANCH TO SINGLE SPACING
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         B     **+8                    BRANCH AROUND SINGLE SPACING
         MVI   ELINE,C' '              SINGLE SPACE ERROR LINE
         MVC   ELINE+27(1),13(2)       MOVE INCORRECT CHAR INTO ERR MSG
         MVC   ELINE+16(4),=C'GOPR'    INSERT GRAD OPR INTO ERR MSG
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE               PRINT GRAD OPR ERROR MSG
         MVI   ELINE,C'0'              DOUBLE SPACE ERROR LINE
         MVI   0(5),C' '               SINGLE SPACE OUTPUT LINE
         MVC   112(3,5),14(8)          MOVE GRAD OPR INTO OUTPUT LINE
         MVC   117(16,5),24(8)         MOVE 1ST DUSTA INTO OUTPUT LINE
         SR    12,10                   SUBTRACT 1 FROM NR OF DUTY STAS
         SR    6,11                    SUBTRACT 2 FROM LINE CTR
         PUT   OUT,LINE                PRINT 1ST LINE OF OUTPUT

*        ********************************************************************
*        ********************************************************************
*        **********                                              **********
*        **********  'LOAD EDIT AND PRINT'ROUTINE FOR LINE       **********
*        **********              TWO DATA OUTPUT                 **********
*        **********                                              **********
*        ********************************************************************
*        ********************************************************************

         MVI   ELINE,C' '              SINGLE SPACE ERROR MSG LINE
         MVI   0(5),C' '               INSERT BLANK CHAR IN OUTPUT LINE
         MVC   1(132,5),0(5)           CLEAR OUTPUT LINE

*        ********************************************************************
*        **********                                              **********
*        **********              RANK CODE CHECK                 **********
*        **********                                              **********
*        ********************************************************************

         CLI   37(7),X'C1'             TEST RANK CDE FOR MIN VALID CHAR
         BC    4,*+12                  COND BRANCH TO RANK ERROR ROUT
         CLI   37(7),X'E3'             TEST RANK CDE FOR MAX VALID CHAR
         BC    13,*+36                 COND BRCH AROUND RANK ERROR ROUT
         MVC   ELINE+27(1),37(7)       MOVE BAD RK CDE CHAR IN ERR MSG
         MVC   ELINE+16(4),=C'RANK'    INSERT RANK INTO ERROR MSG
```

```
*        SR    6,10                     SUBTRACT 1 FROM LINE CTR
*        PUT   OUT,ELINE                PRINT RANK ERROR MSG
*        B     *+72                     BRANCH AROUND RANK DECODING ROUT
*        SR    2,2                      CLEAR R2
         MVC   TEST,37(7)               MOVE RANK CODE INTO TEST AREA
         NI    37(7),X'0F'              CONVERT RANK CODE TO BINARY NR
         IC    2,37(7)                  INSERT BINARY NR IN R2
         SLL   2,2                      MULTIPLY NR BY 4
         TM    TEST,X'30'               TEST RANK CODE FOR A-I
         BC    8,*+28                   COND BRANCH TO A-I RK TABLE SEG
         TM    TEST,X'10'               TEST RANK CODE FOR J-R SEGMENT
         BC    8,*+12                   COND BRANCH TO S-Z RK TABLE SEG
         LA    3,TABLEG+32              SET R3 TO J-R RANK TABLE SEGMENT
         B     *+16                     BRANCH TO RK TABLE POINTER ROUT
         LA    3,TABLEG+64              SET R3 TO S-Z RANK TABLE SEGMENT
         B     *+8                      BRANCH TO RK TABLE POINTER ROUT
         LA    3,TABLEG-4               SET R3 TO A-I RANK TABLE SEGMENT
         AR    3,2                      SET R3 TO CORRECT RANK ADDRESS
         MVC   1(4,5),0(3)              MOVE RANK INTO OUTPUT LINE
         MVI   5(5),C','                INSERT COMMA
         CLI   23(8),C'0'               TEST FOR FOREIGN STUDENT FLAG
         BC    8,CTY                    COND BRANCH AROUND SVCNR ROUT
*        ***************************************************************
*        **********                                           **********
*        ********** SERVICE NUMBER AND FOREIGN COUNTRY CHECK  **********
*        **********                                           **********
*        ***************************************************************
         LR    2,7                      STORE CARD 1 ADDR IN R2
         SR    2,10                     SUBTRACT 1 FROM CARD 1 ADDR
         LA    3,6                      INSERT CONSTANT 6 IN R3
         AR    2,10                     ADD 1 TO CARD 1 ADDR
         CLI   0(2),C'0'                TEST SVCNR FOR VALID CHAR
         BC    4,*+12                   COND BRANCH TO SVCNR ERR ROUT
         BCT   3,*-10                   BRANCH FOR NEXT CHAR TEST
         B     *+32                     BRANCH AROUND SVCNR ERR ROUT
         MVC   ELINE+27(1),0(2)         MOVE INCORRECT CHAR INTO ERR MSG
         MVC   ELINE+16(4),=C'SRNR'     INSERT SRNR IN ERR MSG
         SR    6,10                     SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE                PRINT SVCNR ERROR MSG
         MVC   6(6,5),0(7)              MOVE SERIAL NR INTO OUTPUT LINE
         MVI   12(5),C','               INSERT COMMA
CTY      SR    2,2                      CLEAR R2
         CLI   21(8),X'F0'              TEST VALIDITY OF SVC/CTY CODE
         BC    4,*+22                   COND BRANCH TO CTR ERROR ROUT
         CLI   21(8),X'F2'              TEST VALIDITY OF SVC/CTY CODE
         BC    4,SVC                    COND BRANCH TO SVC DECODING ROUT
```

```
         CLC   21(2,8),=CL2'41'        TEST CTY CODE FOR MAX VALUE
         BC    13,*+40                 COND BRANCH AROUND CTR ERR ROUT
         MVC   ELINE+26(2),21(8)       MOVE BAD CTY CODE INTO ERROR MSG
         MVC   ELINE+16(4),=C'CTY '    INSERT CTY IN ERROR MSG
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE               PRINT CTY ERROR MSG
         MVI   ELINE+26,C' '           CLEAR OLD CODE FROM ERROR MSG
         B     SEX                     BRANCH TO NEXT OUTPUT LINE ENTRY
         NI    21(8),X'0F'             CONVERT 1ST CTY CDE DIGIT TO BIN
         NI    22(8),X'0F'             CONVERT 2ND CTY CDE DIGIT TO BIN
         LA    4,TABLEC-160            SET R4 AS CTY TABLE POINTER
         IC    2,21(8)                 INSERT 1ST CTY CODE DIGIT IN R2
         LA    3,80                    SET R3 TO 80
         AR    4,3                     ADD 10 LINES TO CTY POINTER
         BCT   2,*-2                   INCREASE POINTER VALUE IF REQRD
         IC    2,22(8)                 INSERT 2ND CTY CODE DIGIT IN R2
         SLL   2,3                     MULTIPLY R2 BY 8
         AR    4,2                     SET CTY TABLE ADDR POINTER (R4)
         MVC   13(8,5),0(4)            MOVE CTY NAME INTO OUTPUT LINE
         B     SEX                     BRANCH TO NEXT OUTPUT LINE ENTRY
SVC      CLI   22(8),X'F6'             TEST SVC CODE FOR MAX VALUE
         BC    13,*+40                 COND BRANCH AROUND SVC ERR ROUT
         MVC   ELINE+26(2),21(8)       MOVE BAD SVC CODE INTO ERROR MSG
         MVC   ELINE+16(4),=C'SVC '    INSERT SVC IN ERROR MSG
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE               PRINT SVC ERROR MSG
         MVI   ELINE+26,C' '           CLEAR OLD CODE FROM ERROR MSG
         B     SEX                     BRANCH AROUND SVC DECODING ROUT
         CLC   21(2,8),=C'01'          TEST FOR " IGEP " ENTRY
         BC    7,*+8                   COND BRANCH AROUND IGEP ENTRY
         MVI   30(5),X'C9'             MOVE I INTO OUTPUT LINE
         NI    22(8),X'0F'             CONVERT SVC CODE TO BINARY NR
         IC    2,22(8)                 INSERT NR IN R2
         SLL   2,2                     MULTIPLY NR BY 4
         CLI   21(8),C'0'              TEST FOR NON-US OFFICER
         BC    8,*+12                  COND BRANCH TO USN TABLE
         LA    3,TABLES+12             SET R3 TO NON-USN POINTER
         B     *+8                     BRANCH AROUND USN POINTER
         LA    3,TABLES                SET R3 TO SVC TABLE BASE ADDR
         AR    3,2                     SET SVC TABLE ADDR POINTER (R3)
         MVC   13(4,5),0(3)            MOVE SVC INTO OUTPUT LINE
SEX      CLI   64(7),X'C6'             TEST FOR    ALE SEX INDICATOR
         BC    7,*+8                   COND BRA      OUND SEX ENTRY
         MVI   30(5),X'C6'             MOVE F IN.    PUT LINE
         CLI   0(8),C' '               TEST FOR DESIGNATOR CHANGE ENTRY
         BC    8,*+10                  COND BRCH AROUND OLD DESIG DATA
         MVI   53(5),C'('              INSERT OPENING PAREN
         MVC   54(4,5),0(8)            MOVE DESIG CHG DTE IN OUTPT LINE
```

```
         MVI   58(5),C')'              INSERT CLOSING PAREN
*        ********************************************************************
*        **********                                                **********
*        **********          GRADUATION DATE CHECK                 **********
*        **********                                                **********
*        ********************************************************************
         LR    2,8                     STORE CARD 2 ADDR IN R2
         LA    3,3                     INSERT CONSTANT 3 IN R3
         AR    2,10                    ADD 1 TO CARD 2 ADDR
         CLI   10(2),C'0'              TEST GRD DTE FOR VALID CHAR
         BC    4,*+12                  COND BRANCH TO ERR ROUT
         BCT   3,*-10                  BRANCH FOR NEXT CHAR TEST
         B     *+32                    BRANCH AROUND END DATE ERR ROUT
         MVC   ELINE+27(1),10(2)       MOVE INCORRECT CHAR INTO ERR MSG
         MVC   ELINE+16(4),=C'EDTE'    INSERT EDTE INTO ERR MSG
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE               PRINT END DATE ERROR MSG
         MVC   94(1,5),11(8)           MOVE END QTR INTO OUTPUT LINE
         MVI   95(5),C'-'              INSERT SPACER
         MVC   96(2,5),12(8)           MOVE END YEAR INTO OUTPUT LINE
*        ********************************************************************
*        **********                                                **********
*        **********          DEGREE AREA CHECK                     **********
*        **********                                                **********
*        ********************************************************************
         CLI   20(8),C'A'              TEST AREA CODE FOR MAX VALUE
         BC    4,*+12                  COND BRANCH TO AREA ERROR ROUT
         CLI   20(8),C'T'              TEST AREA CODE FOR MAX VALUE
         BC    13,*+36                 COND BRCH AROUND AREA ERROR ROUT
         MVC   ELINE+27(1),20(8)       MOVE BAD AREA CODE INTO ERR MSG
         MVC   ELINE+16(4),=C'AREA'    INSERT AREA INTO ERROR MSG
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,ELINE               PRINT AREA ERROR MSG
         B     *+62                    BRANCH AROUND AREA DECODING ROUT
         SR    2,2                     CLEAR R2
         MVC   TEST,20(8)              MOVE AREA CODE INTO TEST AREA
         NI    20(8),X'0F'             CONVERT AREA CODE TO BINARY NR
         IC    2,20(8)                 INSERT BINARY NR IN R2
         SLL   2,2                     MULTIPLY NR BY 4
         TM    TEST,X'30'              TEST DEG AREA CDE FOR A-I SEG
         BC    8,*+28                  COND BRANCH TO A-I DEG TABLE SEG
         TM    TEST,X'10'              TEST DEG AREA CDE FOR J-R SEG
         BC    8,*+12                  COND BRANCH TO S-Z DEG TABLE SEG
         LA    3,TABLED+32             SET R3 TO DEG TABLE SEGMENT
         B     *+16                    BRANCH TO TABLE PTR ROUTINE
```

```
LA     3,TABLED+64                 SET R3 TO S-7 DEG TABLE SEGMENT
B      *+8                         BRANCH TO TABLE PTR ROUTINE
LA     3,TABLED-4                  SET R3 TO A-I DEG TABLE SEGMENT
AR     3,2                         SET AREA TABLE PTR TO CODE ADDR
MVC    106(4,5),0(3)               MOVE DEGR AREA INTO OUTPUT LINE

*      *****************************************************************
*      **********                                             **********
*      **********              TOTAL 'OPR' CHECK              **********
*      **********                                             **********
*      *****************************************************************

LR     2,8                         STORE CARD 2 ADDR IN R2
LA     3,3                         INSERT CONSTANT 3 IN R3
AR     2,10                        ADD 1 TO CARD 2 ADDR
CLI    16(2),C'0'                  TEST TOT OPR FOR VALID CHAR
BC     4,*+12                      COND BRANCH TO ERR ROUT
BCT    3,*-10                      BRANCH FOR NEXT CHAR TEST
B      *+32                        BRANCH AROUND ERR ROUT
MVC    ELINE+27(1),16(2)           MOVE INCORRECT CHAR INTO ERR MSG
MVC    ELINE+16(4),=C'TOPR'        INSERT TOPR INTO ERR MSG
SR     6,10                        SUBTRACT 1 FROM LINE CTR
PUT    OUT,ELINE                   PRINT TOTAL OPR ERROR MSG
MVC    112(3,5),17(8)              MOVE TOTAL OPR INTO OUTPUT LINE
MVC    117(16,5),40(8)             MOVE 2ND DUTYSTA INTO OUTPT LINE
SR     12,10                       SUBTRACT 1 FROM NO OF DUTY STAS
SR     6,10                        SUBTRACT ONE FROM PAGE LINE CTR
PUT    OUT,LINE                    PRINT 2ND LINE OF OUTPUT

*      *****************************************************************
*      *****************************************************************
*      **********                                             **********
*      ********** 'LOAD EDIT AND PRINT'ROUTINE FOR LINE       **********
*      **********           THREE DATA OUTPUT                 **********
*      **********                                             **********
*      *****************************************************************
*      *****************************************************************

CLI    36(7),C'*'                  TEST FOR NEW DECK INDICATOR
BC     3,*+20                      COND BRANCH AROUND ERR MSG ROUT
SR     6,10                        SUBTRACT ONE FROM PAGE LINE CTR
PUT    OUT,STARMSG                 PRINT ERROR MSG
CLI    23(8),C'0'                  TEST FOR FOREIGN STUDENT FLAG
BC     8,READ                      COND BRANCH TO READ NEW DECK
MVI    0(5),C' '                   INSERT BLANK CHAR IN OUTPUT LINE
MVC    1(132,5),0(5)               CLEAR OUTPUT LINE
CLI    4(8),C' '                   TEST FOR DESIGNATOR CHG DATA
```

```
         BC    7,*+10                    COND BRANCH TO OLD DESIG ENTRY
         CP    12,10                     TEST NR OF DUTY STAS LEFT TO PRT
         BC    11,*+18                   COND BRANCH TO DUTY STA ENTRY
         MVI   53(5),C'('                INSERT OPENING PAREN
         MVC   54(4,5),4(8)              MOVE OLD DESIG INTO OUTPUT LINE
         MVI   58(5),C')'                INSERT CLOSING PAREN
         MVC   117(16,5),56(8)           MOVE 3RD DUTYSTA INTO OUTPT LINE
         SP    6,10                      SUBTRACT 1 FROM LINE CTR
         PUT   OUT,LINE                  PRINT 3RD LINE OF OUTPUT

*        ********************************************************************
*        ********************************************************************
*        **********                                              **********
*        **********  'LOAD EDIT AND PRINT'ROUTINE FOR LINE       **********
*        **********     FOUR THROUGH EIGHT DATA OUTPUT           **********
*        **********                                              **********
*        ********************************************************************
*        ********************************************************************

         CLI   DBCK,C'*'                 TEST FOR 2 CARD DATA DECK FLAG
         MVI   DBCK,C' '                 CLEAR 2 CARD DATA DECK FLAG
         BC    8,READ                    COND BRANCH TO CARD READ ROUTINE
         SP    12,10                     SUBTRACT 1 FROM NR OF DUTY STAS
         BC    12,READ                   COND BRANCH TO CARD READ ROUTINE
         LR    2,9                       MOVE 3RD DATA CARD ADDR INTO R2
         MVI   0(5),C' '                 INSERT BLANK CHAR IN OUTPUT LINE
         MVC   1(132,5),0(5)             CLEAR OUTPUT LINE
         MVC   117(16,5),0(2)            MOVE NEXT DUTYSTA TO OUTPUT LINE
         SP    6,10                      SUBTRACT 1 FROM LINE CTR
         PUT   OUT,LINE                  PRINT NEXT OUTPUT LINE
         LA    2,16(2)                   SET DUTY STA POINTER TO NEXT STA
         BCT   12,*-36                   COND BRANCH TO DUTSTA PRT ROUT
         B     READ                      BRANCH TO CARD READ ROUTINE

*        ********************************************************************
*        ********************************************************************
*        **********                                              **********
*        **********          PROGRAM CLOSE ROUTINE               **********
*        **********                                              **********
*        ********************************************************************
*        ********************************************************************

MYEOF    PUT   OUT,END                   PRINT END OF DATA MESSAGE
         CLOSE (IN,,OUT)
         L     13,4(13)
         RETURN (14,12),T
```

```
*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********    TABLES,DEFINED CONSTANTS,RESERVED           **********
*        **********  STORAGE AREAS AND INPUT/OUTPUT DATA           **********
*        **********              DATA CONTROL                      **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************

*        ********************************************************************
*        **********                                                **********
*        **********          FOREIGN COUNTRY TABLE                 **********
*        **********                                                **********
*        ********************************************************************

TABLEC   DC      C'BRAZIL   '
         DC      C'CANADA   '
         DC      C'CEYLON   '
         DC      C'CHILE    '
         DC      C'CHINA    '
         DC      C'COLOMBIA '
         DC      C'EQUADOR  '
         DC      C'GERMANY  '
         DC      C'INDIA    '
         DC      C'IRAN     '
         DC      C'ISRAEL   '
         DC      C'KOREA    '
         DC      C'NORWAY   '
         DC      C'PERU     '
         DC      C'PHILL.   '
         DC      C'PORT.    '
         DC      C'THAILAND '
         DC      C'TURKEY   '
         DC      C'URUAGUAY '
         DC      C'VENE.    '
         DC      C'VIETNAM  '
         DC      C'FOREIGN  '

*        ********************************************************************
*        **********                                                **********
*        **********      DEGREE (EDUCATIONAL) AREA TABLE           **********
*        **********                                                **********
*        ********************************************************************

TABLED   DC      C'NONE'
         DC      C'A.E.'
         DC      C'AENG'
```

RANK AND TITLE TABLE
TABLE6

```
*        ******************************************************************
*        **********                                              **********
*        **********          EDUCATIONAL LEVEL TABLE             **********
*        **********                                              **********
*        ******************************************************************

TABLEL   DC    C'GSCH'
         DC    C'HSCH'
         DC    C'1YRC'
         DC    C'2YRC'
         DC    C'3YRC'
         DC    C' PG '
         DC    C'BACH'
         DC    C'CERT'
         DC    C' MS '
         DC    C'PHD '

*        ******************************************************************
*        **********                                              **********
*        **********            ARMED FORCES TABLE                **********
*        **********                                              **********
*        ******************************************************************

TABLES   DC    C'USN '
         DC    C'USN '
         DC    C'USNR'
         DC    C'USA '
         DC    C'USCG'
         DC    C'USMC'
         DC    C'USAF'

*        ******************************************************************
*        **********                                              **********
*        ********** DATA STATUS MESSAGES AND TITLE CONSTANTS     **********
*        **********                                              **********
*        ******************************************************************

TITL1    DC    C'1NAME                                  SOCIAL      BIRTH      *
                    RANK     ORIG          1ST             NPGS     CUR  DEGR*
                                       '
TITL2    DC    C' RANK,FILE NR,SVC/CTY(,*)            SECURITY      DATE   ,   *
               DESIG    DATE    SOURCE  YR      RET    P     QTR-YR  PIC    /   *
                 OPR                    '
TITL3    DC    C' (*=F IF FEMALE,=I IF IGEP)           NUMBER       YR-M-D      *
               (CHG)   YR-M-D    CODE    GP    YR    CODE   ST/END  LUM   AREA*
                 G/T   PAST DUTY STAS   '
```

```
TITL4    DC    C'+________________________________   _________   ______   *
               _____  ______  ______  T  ___  ____  ______  ___  ____*
               ___  _______________
FLINE    DC    C' **** INCORRECT       CODE=   ****                         *
                                       '
STARMSG  DC    C' **** NEW RECORD IND MISSING ****                         *
                                       '
WARNING  DC    C'0********************'       WARNING CARD DECK DELETED*
                DUE TO ERRONEOUS DATA   *******************            *
                                       '
END      DC    C'0##########         END OF DATA            ##########  *
                          END OF DATA       ##########         END OF *
               DATA           #########'
*              ********************************************************
*              **********                                    **********
*              **********          DEFINED STORAGE AREA      **********
*              **********                                    **********
*              ********************************************************
CDAREA   DS    CL240
DBCK     DS    CL1
LINE     DS    CL133
TEST     DS    CL1
*              ********************************************************
*              **********                                    **********
*              **********   INPUT/OUTPUT DATA CONTROL BLOCKS **********
*              **********                                    **********
*              ********************************************************
IN       DCB   DDNAME=MASSAIIN,DSORG=PS,MACRF=(GT),BLKSIZE=80,RECFM=F,E*
               ODAD=MYEOF,BUFNO=3
OUT      DCB   DDNAME=MASSAOUT,DSORG=PS,MACRF=(PM),BLKSIZE=133,RECFM=FA
               LTORG
               END
//GO.MASSAOUT DD SYSOUT=A
//GO.MASSAIIN DD *
```

```
// EXEC ASMALG                          BUPSXFR
//ASM.SYSIN DD *
*********************************************************************
*********************************************************************
*********************************************************************
**                                                                 **
**                                                                 **
** THIS PROGRAM WILL TRANSFER NAVAL BUREAU OF PERSONNEL DATA ON ALL **
** OFFICERS WHO HAVE ATTENDED THE U.S.NAVAL POSTGRADUATE SCHOOL AND **
** ARE STILL ON ACTIVE DUTY IN A FORMAT THAT WILL BE ACCEPTED BY THE **
** STUDENT OFFICER INFORMATION RETRIEVAL SYSTEM. PRIOR TO INITIATING **
** A LOCAL HISTORICAL FILE THE BUPERS TAPE MUST BE CHANGED FROM    **
** SERVICE NUMBER ORDER TO ALPHABETICAL ORDER USING THE SYSTEM SORT/ **
** MERGE PROCEDURE AVAILABLE AT THE PGS COMPUTER CENTER. PROGRAM   **
** PRINTED OUTPUT WILL INCLUDE A LISTING OF ALL RECORDS NOT        **
** TRANSFERED DUE TO PRESENT ATTENDENCE OR ERRONEOUS ATTENDENCE DATA **
**                                                                 **
**                                                                 **
*********************************************************************
*********************************************************************
*********************************************************************

*          *************************************************************
*          *************************************************************
*          **********                                       ************
*          **********  INITIALIZATION AND OPENING ROUTINE   ************
*          **********                                       ************
*          *************************************************************
*          *************************************************************

TPTOTP     CSECT
           USING *,15
           SAVE  (14,12),T,*
           LR    12,13
           CNOP  0,4
           BAL   13,*+76
           DROP  15
           USING *,13
           DS    18F
           ST    13,8(12)
           ST    12,4(13)
           OPEN  (TAPIN,(INPUT),PRTOUT,(OUTPUT),TAPOUT,(OUTPUT))
           LA    2,TPIN                    SET R2 TO TAPE INPUT AREA ADDR
           LA    3,TPOUT                   SET R3 TO TAPE OUTPUT AREA ADDR
           LA    4,8                       SET CONSTANT 8 IN R4
           LA    5,LINE                    SET R5 TO CARD OUTPUT AREA ADDR
           MVI   0(5),C'0'                 DOUBLE SPACE OUTPUT LINE
```

```
*        ****************************************************************
*        ****************************************************************
*        **********                                            **********
*        **********          READ AND TEST INPUT TAPE          **********
*        **********                                            **********
*        ****************************************************************
*        ****************************************************************

READTP   GET   TAPIN,(2)                 READ BUREAU TAPE RECORD
         LR    2,1                       SAVE TAPE INPUT ADDR IN R2
         CLC   110(8,2),=C'MONTEREY'     TEST FOR VALID PGS MONT ENTRY
         BC    7,PRTDATA                 COND BRANCH TO PRINT BAD REC

*        ****************************************************************
*        **********                                            **********
*        **********  DETERMINE CURRICULUM TABLE ENTRY POINT    **********
*        **********                                            **********
*        ****************************************************************

         CLI   126(2),C'4'               TEST FOR 4000 AREA OF CURRTAB
         BC    2,LOW1                    COND BRANCH TO 6000 AREA
         BC    4,HIGH1                   COND BRANCH TO 3000 AREA
         LA    7,12                      SET CTR TO TWELVE 4000 ENTRIES
         LA    6,CURRTAB+40              SET R6 TO TABLE SEARCH ADDR
         AR    6,4                       ADD ONE ENTRY TO TABLE PTR (R6)
         CLC   126(4,2),0(6)             TEST FOR A TABLE ENTRY MATCH
         BC    8,MAJOR                   COND BRANCH IF EQUAL
         BCT   7,*-12                    BRANCH & TEST NEXT TABLE ENTRY
         B     PRTDATA                   BRANCH TO PRINT BAD DATA ENTRY

*        ****************************************************************
*        **********                                            **********
*        **********    SEARCH CURRICULUM TABLE FOR BUPERS      **********
*        **********    EDUCATION MAJOR AND SPECIALTY ENTRY     **********
*        **********                                            **********
*        ****************************************************************

HIGH1    LA    7,6                       SET CTR TO SIX 3000 ENTRIES
         LA    6,CURRTAB-8               SET R6 TO TABLE SEARCH START ADD
         AR    6,4                       ADD ONE ENTRY TO TABLE PTR (R6)
         CLC   126(4,2),0(6)             TEST FOR A TABLE ENTRY MATCH
         BC    8,MAJOR                   COND BRANCH IF EQUAL
         BCT   7,*-12                    BRANCH & TEST NEXT TABLE ENTRY
         B     PRTDATA                   BRANCH TO PRINT BAD DATA ENTRY
LOW1     CLI   126(2),C'6'               TEST FOR 6000 AREA OF CURRTAB
         BC    2,LOW2                    COND BRANCH TO 8000 AREA
         BC    4,HIGH2                   COND BRANCH TO 5000 AREA
         LA    7,25                      SET CTR TO 25 6000 ENTRIES
```

```
                LA      6,CURRTAB+344           SET R6 TO TABLE SEARCH ADDR
                AR      6,4                     ADD ONE ENTRY TO TABLE PTR (R6)
                CLC     126(4,2),0(6)           TEST FOR A TABLE ENTRY MATCH
                BC      8,MAJOR                 COND BRANCH IF EQUAL
                BCT     7,*-12                  BRANCH & TEST NEXT TABLE ENTRY
                B       PRTDATA                 BRANCH TO PRINT BAD DATA ENTRY
HIGH2           LA      7,26                    SET CTR TO 26 5000 ENTRIES
                LA      6,CURRTAB+136           SET R6 TO TABLE SEARCH ADDR
                AR      6,4                     ADD ONE ENTRY TO TABLE PTR (R6)
                CLC     126(4,2),0(6)           TEST FOR A TABLE ENTRY MATCH
                BC      8,MAJOR                 COND BRANCH IF EQUAL
                BCT     7,*-12                  BRANCH & TEST NEXT TABLE ENTRY
                B       PRTDATA                 BRANCH TO PRINT BAD DATA ENTRY
LOW2            LA      7,10                    SET CTR TO TEN 8000 ENTRIES
                LA      6,CURRTAB+544           SET R6 TO TABLE SEARCH ADDR
                AR      6,4                     ADD ONE ENTRY TO TABLE PTR (R6)
                CLC     126(4,2),0(6)           TEST FOR A TABLE ENTRY MATCH
                BC      8,MAJOR                 COND BRANCH IF EQUAL
                BCT     7,*-12                  BRANCH & TEST NEXT TABLE ENTRY

*               ********************************************************************
*               ********************************************************************
*               **********                                          **********
*               **********  PRINT RECORDS NOT NEEDED FOR LOCAL FILE  **********
*               **********                                          **********
*               ********************************************************************
*               ********************************************************************

PRTDATA         MVC     1(130,5),0(2)           MOVE BAD DATA REC TO PRINT BUFF
                PUT     PRTOUT,LIN              PRINT BAD DATA RECORD
                B       READTP                  BRANCH & READ NEXT RECORD

*               ********************************************************************
*               **********                                          **********
*               **********          EDIT BUPERS DATA RECORDS        **********
*               **********             TO LOCAL FORMAT              **********
*               **********                                          **********
*               ********************************************************************

MAJOR           MVC     76(3,3),4(6)            MOVE CURRICULUM NR TO NEW RECORD
                MVC     0(6,3),31(2)            MOVE SERVICE NR TO NEW RECORD
                MVC     6(31,3),0(2)            MOVE NAME TO NEW RECORD
                MVC     37(11,3),46(2)          MOVE RANK,DESIG,YR GP & S-CODE
                MVI     48(3),C' '              INSERT BLANK IN 1ST DOR COL
                MVC     49(5,3),48(3)           FILL DATE OF RANK FIELD WITH BLK
                MVC     54(4,3),61(2)           MOVE PCODE TO NEW RECORD
                MVC     58(6,3),87(2)           MOVE DTE OF BIRTH TO NEW RECORD
```

```
         CLI   93(2),C'W'              TEST FOR FEMALE ENTRY
         BC    7,*+12                  COND BRANCH AROUND "F" ENTRY
         MVI   64(3),C'F'              MOVE F INTO NEW RECORD
         B     *+8                     BRANCH AROUND NON-FEMALE FILLER
         MVI   64(3),C' '              CLEAR FEMALE ENTRY IN WORK AREA
         MVC   65(9,3),37(2)           MOVE SOC SEC NR TO NEW RECORD
         MVC   74(2,3),57(2)           MOVE 1ST RET YR TO NEW RECORD
         MVC   79(1,3),125(2)          MOVE EDUCATION LEVEL TO RECORD
         MVC   80(8,3),94(2)           MOVE OLD DESIGNATOR DATA
         MVI   88(3),C' '              INSERT BLANK CHARACTER
         MVC   89(11,3),88(3)          CLEAR OPR & GRAD DATE FIELD
         MVC   92(2,3),122(2)          MOVE SCH GRAD YR TO NEW RECORD
         MVC   100(1,3),7(6)           MOVE DEGREE AREA TO NEW RECORD
         MVC   101(3,3),=C'001'        SET SVC CDE TO NAVY & DUSTA TO 1
         MVC   104(16,3),71(2)         MOVE LAST DUTY STA TO NEW REC
         MVI   120(3),C' '             INSERT BLANK CHAR IN TAPE FIELD
         MVC   121(112,3),120(3)       CLEAR UNUSED TAPE FIELD

*        ********************************************************************
*        ********************************************************************
*        **********                                              **********
*        **********      WRITE LOCAL TAPE FILE RECORDS           **********
*        **********                                              **********
*        ********************************************************************
*        ********************************************************************

         PUT   TAPOUT,TPOUT            PRINT NEW TAPE RECORD
         B     READTP                  BRANCH & READ NEXT RECORD

*        ********************************************************************
*        **********                                              **********
*        **********          PROGRAM CLOSE ROUTINE               **********
*        **********                                              **********
*        ********************************************************************

MYEOF    PUT   PRTOUT,END              PRINT END OF DATA MESSAGE
         CLOSE (TAPIN,,PRTOUT,,TAPOUT)
         L     13,4(13)
         RETURN (14,12),T
```

```
*          ******************************************************************
*          ******************************************************************
*          **********                                              **********
*          **********     TABLES,DEFINED CONSTANTS,RESERVED        **********
*          **********    STORAGE AREAS AND INPUT/OUTPUT DATA       **********
*          **********              DATA CONTROL                    **********
*          **********                                              **********
*          ******************************************************************
*          ******************************************************************

*          ******************************************************************
*          **********                                              **********
*          **********       CURRICULUM CONVERSION TABLE            **********
*          **********                                              **********
*          ******************************************************************
CURRTAB    DC     C'0000461A'
           DC     C'3500461A'
           DC     C'3600360J'
           DC     C'3700372H'
           DC     C'3701372H'
           DC     C'3800380R'
           DC     C'4100461A'
           DC     C'4200380S'
           DC     C'4300430P'
           DC     C'4301380P'
           DC     C'4400380Q'
           DC     C'4414380Q'
           DC     C'4432380Q'
           DC     C'4438440I'
           DC     C'4460440I'
           DC     C'4463371D'
           DC     C'4600460A'
           DC     C'4615368L'
           DC     C'5212521Q'
           DC     C'5300530F'
           DC     C'5303530F'
           DC     C'5308530F'
           DC     C'5309530R'
           DC     C'5311530Q'
           DC     C'5314521Q'
           DC     C'5319530Q'
           DC     C'5330530Q'
           DC     C'5334530S'
           DC     C'5335530F'
           DC     C'5336530R'
           DC     C'5339535Q'
```

```
         DC    C'5600570F'
         DC    C'5607610B'
         DC    C'5700570N'
         DC    C'5701570N'
         DC    C'5702570N'
         DC    C'5714570N'
         DC    C'5725570N'
         DC    C'5727570N'
         DC    C'5734570N'
         DC    C'5900590F'
         DC    C'5940590F'
         DC    C'5950472F'
         DC    C'5977590F'
         DC    C'6000600F'
         DC    C'6005620A'
         DC    C'6100610B'
         DC    C'6106610B'
         DC    C'6108610B'
         DC    C'6111610B'
         DC    C'6114610B'
         DC    C'6119610B'
         DC    C'6120610B'
         DC    C'6121610B'
         DC    C'6122610B'
         DC    C'6123610B'
         DC    C'6125610B'
         DC    C'6129610B'
         DC    C'6130610B'
         DC    C'6131610B'
         DC    C'6132610B'
         DC    C'6133610B'
         DC    C'6136610B'
         DC    C'6139610B'
         DC    C'6153610B'
         DC    C'6177610B'
         DC    C'6600600T'
         DC    C'6700461E'
         DC    C'6800461E'
         DC    C'8100817M'
         DC    C'8116817M'
         DC    C'8134817M'
         DC    C'8151817M'
         DC    C'8158817M'
         DC    C'8159367K'
         DC    C'8171817M'
         DC    C'8173817M'
         DC    C'8181817M'
         DC    C'8500817M'
```

```
*          ***********************************************************
*          **********                                       **********
*          **********          END OF DATA MESSAGE          **********
*          **********                                       **********
*          ***********************************************************
END        DC    C'0##########                 END OF DATA        ########## *
                              END OF DATA          ##########     END OF *
                 DATA                   #########'
*          ***********************************************************
*          **********                                       **********
*          **********          DEFINED STORAGE AREA         **********
*          **********                                       **********
*          ***********************************************************
LINE       DS    CL133                    OUTPUT LINE PRINT AREA
TPIN       DS    CL150                    TAPE READ AREA
TPOUT      DS    CL232                    TAPE WRITE AREA
*          ***********************************************************
*          **********                                       **********
*          **********   INPUT/OUTPUT DATA CONTROL BLOCKS    **********
*          **********                                       **********
*          ***********************************************************
TAPIN      DCB   DDNAME=MASRTAPE,DSORG=PS,MACRF=(GT),LRECL=150,BLKSIZE=15*
                 00,RECFM=FB,EODAD=MYEOF
TAPOUT     DCB   DDNAME=MASTPOUT,DSORG=PS,MACRF=(PM),LRECL=232,BLKSIZE=92*
                 8,RECFM=FB,EODAD=MYEOF
PRTOUT     DCB   DDNAME=MASPTOUT,DSORG=PS,MACRF=(PM),BLKSIZE=133,RECFM=FA
           LTORG
           END
//GO.MASRTAPE DD DSNAME=HEN.SORT,UNIT=2400,VOLUME=SER=NPS209,DISP=(OLD,
//          KEEP),LABEL=(1,SL)
//GO.MASPTOUT DD SYSOUT=A
//GO.MASTPOUT DD DSNAME=HEN.MAS,UNIT=2400,VOLUME=SER=NPS138,DISP=(NEW,
//          KEEP,DELETE),LABEL=(1,SL)
```

```
// EXEC ASMALG
//ASM.SYSIN DD *                    0' DATE

*********************************************************************************
*********************************************************************************
*********************************************************************************
**                                                                             **
**                                                                             **
** THIS PROGRAM WILL ADD, DELETE, UPDATE AND TRANSFER INDIVIDUAL               **
** RECORDS CONTAINED IN THE CURRENT TAPE FILE OF THE STUDENT OFFICER           **
** INFORMATION RETRIEVAL SYSTEM. ALL RECORDS AFFECTED ARE PRINTED IN           **
** THEIR ORIGINAL AND CHANGED VERSIONS. INPUT DATA CARDS TO DELETE,            **
** UPDATE AND TRANSFER RECORDS MUST BE IN ALPHABETICAL ORDER. ERROR            **
** MESSAGES WILL OCCUR IF DATA CARDS ARE OUT OF SEQUENCE OR CONTAIN            **
** INCORRECT NAME/SOCIAL SECURITY NUMBERS AND PROGRAM WILL CONTINUE            **
** TO PROCESS INPUT DATA CARDS. A SYSTEM SORT/MERGE PROCEDURE MUST             **
** BE USED ON NEWLY CREATED CURRENT AND HISTORICAL "SON" FILES                 **
** BEFORE USING THE TAPE SEARCH PROGRAM.                                       **
**                                                                             **
**                                                                             **
*********************************************************************************
*********************************************************************************
*********************************************************************************

*        *************************************************************
*        *************************************************************
*        **********                                         **********
*        **********  INITIALIZATION AND OPENING ROUTINE     **********
*        **********                                         **********
*        *************************************************************
*        *************************************************************

FILUPDT  CSECT
         USING *,15
         SAVE  (14,12),T,*
         LR    12,13
         CNOP  0,4
         BAL   13,*+76
         DROP  15
         USING *,13
         DS    18F
         ST    13,8(12)
         ST    12,4(13)
         OPEN  (CDIN,(INPUT),PRTOUT,(OUTPUT))
         OPEN  (TPAIN,(INPUT),TPAOUT,(OUTPUT))
         OPEN  (TPBOUT,(OUTPUT))
         LA    2,CDAREA                    SET R2 TO CARD INPUT AREA ADDR
```

```
                LA    3,TPAREA                  SET R3 TO TAPE INPUT AREA ADDR
                LA    4,LINE                    SET R4 TO PRINT OUTPUT AREA ADDR
                PUT   PRTOUT,START              PRINT START OF DATA MESSAGE

*               ********************************************************************
*               ********************************************************************
*               **********                                          **********
*               **********          READ DATA CARD ROUTINE          **********
*               **********                                          **********
*               ********************************************************************
*               ********************************************************************

READCD          GET   CDIN,(2)                  READ DATA CARD
                LR    2,1                       SAVE CARD INPUT ADDR IN R2
                CLI   36(2),C'*'                TEST FOR NEW CURRENT FILE DATA
                BC    8,ADDREC                  COND BRANCH TO NEW DATA ROUTINE
                CLI   0(2),C'*'                 TEST FOR FOREIGN STUDENT
                BC    8,READTP2-6               COND BRANCH TO ALPHA SEARCH ROUT
                MVC   SSNMASK(9),19(2)          SET WANTED SOCSECNR IN TEST MASK

*               ********************************************************************
*               ********************************************************************
*               **********                                          **********
*               **********   READ OLD CURRENT TAPE FILE ROUTINES    **********
*               **********                                          **********
*               ********************************************************************
*               ********************************************************************

READTP1         GET   TPAIN,(3)                 READ CURRENT TAPE RECORD
                LR    3,1                       SAVE TAPE INPUT ADDR IN R3
                CLC   SSNMASK(9),65(3)          TEST TAPE REC FOR CURRECT SSNR
                BC    8,PRINT                   COND BRANCH TO UPDATE ROUT IF EQ
                PUT   TPAOUT,TPAREA             RECORD GOOD DATA CURRENT TAPE
                B     READTP1                   BRANCH AND READ NEXT TAPE RECORD
                MVC   NMEMASK,1(2)              SET WANTED NAME IN TEST MASK
READTP2         GET   TPAIN,(3)                 READ CURRENT TAPE RECORD
                LR    3,1                       SAVE TAPE INPUT ADDR IN R3
                CLC   NMEMASK,6(3)              TEST TAPE REC FOR CORRECT NAME
                BC    8,PRINT                   COND BRANCH TO UPDATE ROUT IF EQ
                PUT   TPAOUT,TPAREA             RECORD GOOD DATA CURRENT TAPE
                B     READTP2                   BRANCH AND READ NEXT TAPE RECORD
```

```
*        ****************************************************************
*        **********                                            **********
*        **********    TEST FOR RECORD DELETION SYMBOL AND     **********
*        **********       PRINT OLD CURRENT FILE RECORD        **********
*        **********                                            **********
*        ****************************************************************

PRINT    MVI   0(4),C'0'                DOUBLE SPACE OUTPUT LINE
         MVC   1(132,4),0(3)            MOVE OLD RECORD TO PRINT AREA
         CLI   79(2),C'*'               TEST FOR RECORD DELETION SYMBOL
         BC    7,UPDATE                 COND BRANCH TO UPDATE RECORD
         MVC   122(10,4),=C'* DELETE *' INSERT DELETE DATA SYMBOL
         PUT   PRTOUT,LINE              PRINT DELETED DATA RECORD
         B     READCD                   BRANCH AND READ NEW DATA CARD

*        ****************************************************************
*        **********                                            **********
*        **********       UPDATE FILE RECORD ROUTINE           **********
*        **********        AND PRINT ORIGINAL RECORD           **********
*        **********                                            **********
*        ****************************************************************

UPDATE   PUT   PRTOUT,LINE              PRINT ORIGINAL DATA RECORD
         CLI   29(2),C' '               TEST FOR DESIG UPDATE
         BC    8,*+22                   COND BRANCH TO NEXT TEST
         MVC   84(4,3),38(3)            MOVE CHANGED DESIG TO OLD DESIG
         MVC   38(4,3),29(2)            INSERT NEW DESIG
         MVC   80(4,3),33(2)            INSERT DESIG CHANGE DATE
         CLI   39(2),C' '               TEST FOR RANK UPDATE
         BC    8,*+16                   COND BRANCH TO NEXT TEST
         MVC   37(1,3),39(2)            INSERT NEW RANK CODE
         MVC   48(6,3),40(2)            INSERT NEW DATE OF RANK
         CLI   49(2),C' '               TEST FOR CURR UPDATE
         BC    8,*+10                   COND BRANCH TO NEXT TEST
         MVC   76(3,3),49(2)            INSERT NEW CURRICULUM
         CLI   53(2),C' '               TEST FOR EDUCATION AREA UPDATE
         BC    8,*+10                   COND BRANCH TO NEXT TEST
         MVC   100(1,3),53(2)           INSERT NEW EDUCATIONAL CODE
         CLI   55(2),C' '               TEST FOR DEGREE UPDATE
         BC    8,*+10                   COND BRANCH TO NEXT TEST
         MVC   79(1,3),55(2)            INSERT NEW DEGREE CODE
         CLI   59(2),C' '               TEST FOR QPR UPDATE
         BC    8,*+10                   COND BRANCH TO NEXT TEST
         MVC   94(6,3),59(2)            INSERT NEW GRAD & TOTAL QPR
         CLI   69(2),C' '               TEST FOR GRADUATION DATE UPDATE
```

```
         BC    8,*+10                    COND BRANCH AROUND DATE UPDATE
         MVC   91(3,3),69(2)             INSERT NEW GRADUATION DATE

*        ********************************************************************
*        **********                                                **********
*        **********      TEST UPDATED CURRENT FILE RECORD          **********
*        **********        FOR HISTORICAL FILE TRANSFER            **********
*        **********                                                **********
*        ********************************************************************

         MVI   0(4),C' '                 INSERT BLANK IN OUTPUT AREA
         MVC   1(132,4),0(3)             MOVE NEW RECORD TO PRINT AREA
         CLI   79(2),C'$'                TEST FOR DATA TRANSFER SYMBOL
         BC    8,HISTWRT                 COND BRANCH TO HIST TAPE WRITE

*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********         WRITE UPDATED RECORD ON NEW            **********
*        **********         CURRENT OR HISTORICAL FILE             **********
*        **********          AND PRINT UPDATED RECORD              **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************

CURRWRT  PUT   TPAOUT,TPAREA             WRITE NEW CURRENT FILE RECORD
         MVC   122(10,4),=C'* UPDATE *'  INSERT UPDATE DATA INDICATOR
         PUT   PRTOUT,LINE               PRINT UPDATED DATA RECORD
         B     READCD                    BRANCH AND READ NEW DATA CARD
HISTWRT  PUT   TPBOUT,TPAREA             WRITE NEW HISTORICAL FILE RECORD
         MVC   124(8,4),=C'* XFER *'     INSERT TRANSFER DATA INDICATOR
         PUT   PRTOUT,LINE               PRINT TRANSFERED DATA RECORD
         B     READCD                    BRANCH AND READ NEW DATA CARD

*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********  READ NEW DATA CARD DECK AND PRINT NEW         **********
*        **********           CURRENT FILE RECORD                  **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************

ADDREC   MVI   36(2),C' '                INSERT BLANK OVER ADD DECK SYM
         MVC   0(80,3),0(2)              MOVE CARD 1 TO CURRENT TPE OUTPT
         GET   CDIN,(2)                  READ CARD 2
```

```
         LR    2,1                      SAVE CARD INPUT ADDR IN R2
         MVC   80(80,3),0(2)            MOVE CARD 2 TO CURRENT TPE OUTPT
         GET   CDIN,(2)                 READ CARD 3
         LR    2,1                      SAVE CARD INPUT ADDR IN R2
         MVC   160(80,3),0(2)           MOVE CARD 3 TO CURRENT TPE OUTPT
         MVI   0(4),C'0'                INSERT DOUBLE SPACE CHAR
         MVC   1(125,4),0(3)            MOVE NEW RECORD TO PRINT AREA
         MVC   125(7,4),=C'* NEW *'     INSERT NEW DATA INDICATOR
         PUT   PRTOUT,LINE              PRINT NEW DATA RECORD
         PUT   TPAOUT,TPAREA            WRITE NEW CURRENT FILE RECORD
         B     READCD                   BRANCH AND READ NEW DATA CARD

*        ********************************************************************
*        **********                                              **********
*        **********          DATA CARD ERROR ROUTINE             **********
*        **********                                              **********
*        ********************************************************************

MYEOFT1  MVC   ERRMSG+77(19),0(2)       INSERT BAD DATA CARD NME IN ERR
         PUT   PRTOUT,ERRMSG            PRINT ERROR MESSAGE
         CLOSE (TPAIN,REREAD)
         OPEN  (TPAIN,(INPUT))
         B     READCD                   BRANCH TO READ NEXT CARD

*        ********************************************************************
*        **********                                              **********
*        **********          PROGRAM CLOSE ROUTINE               **********
*        **********                                              **********
*        ********************************************************************

MYEOFCD  PUT   PRTOUT,END                PRINT END OF DATA MESSAGE
         CLOSE (CDIN,,TPAIN,,TPAOUT,,TPBOUT,,PRTOUT)
         L     13,4(13)
         RETURN (14,12),T

*        ********************************************************************
*        **********                                              **********
*        **********           DATA STATUS MESSAGES               **********
*        **********                                              **********
*        ********************************************************************

START    DC    C'1##########        START OF DATA          ########## *
                     START OF DATA          ##########          START OF*
                      DATA          ##########'

END      DC    C'0##########          END OF DATA          ########## *
                     END OF DATA            ##########          END OF *
                     DATA          ##########'
```

```
ERRMSG   DC    C'0**********      PROGRAM WILL NOT RUN      **********   *
                DATA UPDATE CARD FOR                        OUT OF SEQUE*
               NCE OR INCORRECT   ****'

*        ************************************************************
*        **********                                        **********
*        **********           DEFINED STORAGE AREA         **********
*        **********                                        **********
*        ************************************************************

LINE     DS    CL133                    PRINT OUTPUT BUFFER AREA
CDAREA   DS    CL80                     CARD INPUT AREA
TPAREA   DS    CL232                    TAPE INPUT AREA
NMEMASK  DS    CL28                     NON-NAVY ALPHA TEST MASK
SSNMASK  DS    CL9                      NAVY SOC SEC NR TEST MASK

*        ************************************************************
*        **********                                        **********
*        **********     INPUT/OUTPUT DATA CONTROL BLOCKS   **********
*        **********                                        **********
*        ************************************************************

CDIN     DCB   DDNAME=MASSAIIN,DSORG=PS,MACRF=(GT),BLKSIZE=80,RECFM=F,E*
               ODAD=MYEOFCD
TPAIN    DCB   DDNAME=MASTPIN1,DSORG=PS,MACRF=(GT),LRECL=232,BLKSIZE=92*
               8,RECFM=FB,EODAD=MYEOFTI
TPAOUT   DCB   DDNAME=MASTOUT1,DSORG=PS,MACRF=(PM),LRECL=232,BLKSIZE=92*
               8,RECFM=FB
TPBOUT   DCB   DDNAME=MASTOUT2,DSORG=PS,MACRF=(PM),LRECL=232,BLKSIZE=92*
               8,RECFM=FB
PRTOUT   DCB   DDNAME=MASPOUT1,DSORG=PS,MACRF=(PM),BLKSIZE=133,RECFM=FA

         LTORG
         END

//GO.MASTPIN1 DD DSNAME=HEN.DONE,UNIT=2400,VOLUME=SER=NPS194,DISP=(OLD,
//         KEEP),LABEL=(1,SL)
//GO.MASTOUT1 DD DSNAME=HEN.CSON,UNIT=2400,VOLUME=SER=NPS155,DISP=(NEW,
//         KEEP),LABEL=(1,SL)
//GO.MASTOUT2 DD DSNAME=HEN.HSON,UNIT=2400,VOLUME=SER=NPS150,DISP=(NEW,
//         KEEP),LABEL=(1,SL)
//GO.MASPOUT1 DD SYSOUT=A
//GO.MASSAIIN DD *
```

```
// EXEC ASMALG
//ASM.SYSIN DD *
*********************************************************************
*********************************************************************
**                                                                 **
**                                                                 **
** THIS PROGRAM WILL SEARCH THE CURRENT OR HISTORICAL TAPE FILES OF **
** THE STUDENT OFFICER INFORMATION RETRIEVAL SYSTEM AND PRINT ALL   **
** SELECTED BY USING COMBINATIONS OF FOURTEEN KEY DATA ITEMS AND    **
** CONDUCTING ANY COMBINATION OF EQUAL, NOT-EQUAL OR GREATER-THAN/  **
** LESS-THAN TESTS ON THE SPECIFIED DATA ITEM FIELDS.              **
**                                                                 **
*********************************************************************
*********************************************************************

*        ***************************************************************
*        ***************************************************************
*        **********                                         **********
*        **********  INITIALIZATION AND OPENING ROUTINE     **********
*        **********                                         **********
*        ***************************************************************
*        ***************************************************************

TPRDPRT  CSECT
         USING *,15
         SAVE  (14,12),T,*
         LR    12,13
         CNOP  0,4
         BAL   13,**+76
         DROP  15
         USING *,13
         DS    18F
         ST    13,8(12)
         ST    12,4(13)
         LA    11,MYEOF1
         USING MYEOF1,11                  SET R11 TO 2ND BASE REG ADDR
         OPEN (IN,(INPUT),RTP,(INPUT),OUT,(OUTPUT))  INITIALIZE 2ND BASE REG
         LA    5,20                       SET R5 TO MAX LOOPS (20)
         LA    3,SSNRCHK                  SET 1ST COMPAR BRANCH ADDR IN R3
         LA    7,CDAREA-80                SET R7 TO CARD READ AREA ADDR-80
         LA    10,1                       LOAD CONSTANT 1 IN R 10
         SR    4,4                        CLEAR R4 (LOOP CTR)
```

```
*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********       READ DATA CARD AND FIND FIRST            **********
*        **********            NON-BLANK CHARACTER                 **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************

READCD   LA      12,80                  SET COLS LEFT TO TEST TO 80 R12
         AR      7,12                   SET R7 TO CARD READ AREA ADDR
         GET     IN,(7)                 READ 1ST DATA CARD
         LR      7,1                    SET R7 TO 1ST COL IN CARD
         SR      2,2                    CLEAR R2
BLKCHK   AR      7,2                    MOVE CARD PTR TO CORRECT COL
         SR      12,2                   SUBTRACT COLUMNS TESTED
         BC      13,READCD              READ NEW CARD IF 80 COLS TESTED
         CLI     0(7),C' '              TEST CARD COL FOR BLANK CHAR
         BC      7,0(3)                 COND BRANCH TO NEXT COMPARE ROUT
         LA      2,1                    SET R2 TO CONSTANT 1
         B       BLKCHK                 BRANCH TO NEXT COL CHECK

*        ********************************************************************
*        **********                                                **********
*        **********       DATA CARD ERROR CHECK ROUTINE            **********
*        **********                                                **********
*        ********************************************************************

SSNRCHK  AR      4,10                   ADD 1 TO LOOP CTR
         CR      5,4                    TEST FOR 20 LOOPS
         BC      4,ERROR                COND BRANCH TO ERROR ROUT IF >20

*        ********************************************************************
*        **********                                                **********
*        **********  SOCIAL SECURITY NUMBER INDICATOR (A)          **********
*        **********            TEST ROUTINE                        **********
*        **********                                                **********
*        ********************************************************************

         CLI     0(7),C'A'              TEST CARD COL FOR SSNR INDICATOR
         BC      7,SEXCHK               COND BRANCH TO NEXT TEST IF NE
         LA      3,SEXCHK               SAVE NEXT TEST ROUT ADDR IN R3
         OI      FLGMASK,X'01'          SET ROUT INDICATOR TO = COMPARE
         MVC     EQMASK+28(9),2(7)      MOVE SSNR INTO = MASK AREA
         MVC     EQMASK+95(9),ONES      MOVE 1'S INTO = AND MASK
         LA      2,12                   ADD 12 TO CARD COL PTR
         B       BLKCHK                 BRANCH TO BLANK CARD COL TEST
```

```
*        ************************************************************
*        **********                                        **********
*        **********       SEX INDICATOR (B) TEST ROUTINE   **********
*        **********                                        **********
*        ************************************************************

SEXCHK   CLI   0(7),C'B'                 TEST CARD COL FOR SEX INDICATOR
         BC    7,PCDECHK                 COND BRANCH TO NEXT TEST IF NE
         LA    3,PCDECHK                 SAVE NEXT TEST ROUT ADDR IN R3
         OI    FLGMASK,X'01'             SET ROUT INDICATOR TO = COMPARE
         MVI   EQMASK+27,C'F'            MOVE F INTO = MASK AREA
         MVI   EQMASK+94,X'FF'           MOVE 1'S INTO = AND MASK
         LA    2,4                       ADD 4 TO CARD COL PTR
         B     BLKCHK                    BRANCH TO BLANK CARD COL TEST

*        ************************************************************
*        **********                                        **********
*        **********    P-CODE INDICATOR (C) TEST ROUTINE   **********
*        **********                                        **********
*        ************************************************************

PCDECHK  CLI   0(7),C'C'                 TEST CARD COL FOR PCODE IND
         BC    7,DESGCHK                 COND BRANCH TO NEXT TEST IF NE
         LA    3,DESGCHK                 SAVE NEXT TEST ROUT ADDR IN R3
         CLI   1(7),C'='                 TEST FOR EQUAL COMPARE INDICATOR
         BC    7,*+24                    COND BRANCH AROUND EQUAL MASKING
         OI    FLGMASK,X'01'             SET ROUT INDICATOR TO = COMPARE
         MVC   EQMASK+17(4),2(7)         MOVE PCODE INTO = MASK AREA
         MVC   EQMASK+84(4),ONES         MOVE 1'S INTO = AND MASK
         B     *+20                      COND BRANCH AROUND UNEQUAL MASK
         OI    FLGMASK,X'02'             SET ROUT IND TO UNEQUAL MASK
         MVC   NEMASK+17(4),2(7)         MOVE PCODE INTO NE MASK AREA
         MVC   NEMASK+84(4),ONES         MOVE 1'S INTO NE AND MASK
         LA    2,7                       ADD 7 TO CARD COL PTR
         B     BLKCHK                    BRANCH TO BLANK CARD COL TEST

*        ************************************************************
*        **********                                        **********
*        **********  DESIGNATOR INDICATOR (D) TEST ROUTINE **********
*        **********                                        **********
*        ************************************************************

DESGCHK  CLI   0(7),C'D'                 TEST CARD COL FOR DESIGNATOR IND
         BC    7,SVCHK                   COND BRANCH TO NEXT TEST IF NE
         LA    3,SVCHK                   SAVE NEXT TEST ROUT ADDR IN R3
         CLI   1(7),C'='                 TEST FOR EQUAL COMPARE INDICATOR
         BC    7,*+24                    COND BRANCH AROUND EQUAL MASKING
         OI    FLGMASK,X'01'             SET ROUT INDICATOR TO = COMPARE
```

```
         MVC   EQMASK+1(4),2(7)        MOVE DESIGNATOR INTO = MASK AREA
         MVC   EQMASK+68(4),ONES       MOVE 1'S INTO = AND MASK
         B     *+20                    COND BRANCH AROUND UNEQUAL MASK
         OI    FLGMASK,X'02'           SET ROUT IND TO UNEQUAL MASK
         MVC   NEMASK+1(4),2(7)        MOVE DESIGNATOR INTO # MASK AREA
         MVC   NEMASK+68(4),ONES       MOVE 1'S INTO # AND MASK
         LA    2,7                     ADD 7 TO CARD COL PTR
         B     BLKCHK                  BRANCH TO BLANK CARD COL TEST

*        ********************************************************************
*        **********                                              **********
*        **********     BRANCH OF SERVICE AND/OR FOREIGN         **********
*        **********    COUNTRY INDICATOR (F) TEST ROUTINE        **********
*        **********                                              **********
*        ********************************************************************

SVCHK    CLI   0(7),C'E'               TEST CARD COL FOR SVC/CNTRY IND
         BC    7,DORCHK                COND BRANCH  TO NEXT TEST IF NE
         LA    3,DORBCHK               SAVE NEXT TEST ROUT ADDR IN R3
         LA    2,4                     ADD 4 TO CARD COL PTR
         CLI   1(7),C'='               TEST FOR EQUAL COMPARE INDICATOR
         BC    7,*+16                  COND BRANCH AROUND EQUAL MASKING
         OI    FLGMASK,X'01'           SET ROUT INDICATOR TO = COMPARE
         LA    6,EQMASK                SAVE EQUAL MASK ADDR IN R6
         B     *+12                    COND BRANCH AROUND UNEQUAL MASK
         OI    FLGMASK,X'02'           SET ROUT IND TO UNEQUAL MASK
         LA    6,NEMASK                SAVE UNEQUAL MASK ADDR IN R6
         CLI   2(7),C'N'               TEST FOR NAVY OFFICERS ONLY
         BC    7,*+16                  COND BRANCH AROUND NAVY MASK
         MVI   64(6),C'0'              MOVE NAVY CODE INTO MASK AREA
         MVI   131(6),X'FF'            MOVE 1'S INTO AND MASK
         B     BLKCHK                  BRANCH TO BLANK CARD COL TEST
         CLI   2(7),C'U'               TEST FOR ALL US OFFICERS
         BC    7,*+16                  COND BRANCH AROUND ALL OFF MASK
         MVI   64(6),C'1'              MOVE ALL OFF CODE INTO MASK AREA
         MVI   131(6),X'FF'            MOVE 1'S INTO AND MASK
         B     BLKCHK                  BRANCH TO BLANK CARD COL TEST
         CLI   2(7),C'F'               TEST FOR FOREIGN OFFICERS
         BC    7,*+16                  COND BRANCH AROUND FOREIGN MASK
         MVI   66(6),C'0'              MOVE FOREIGN CODE INTO MASK AREA
         MVI   133(6),X'FF'            MOVE 1'S INTO AND MASK
         B     BLKCHK                  BRANCH TO BLANK CARD COL TEST
         MVC   64(2,6),2(7)            MOVE SVC/CNTRY CODE TO MASK AREA
         MVC   131(2,6),ONES           MOVE 1'S INTO AND MASK
         LA    2,5                     ADD 5 TO CARD COL PTR
         B     BLKCHK                  BRANCH TO BLANK CARD COL TEST
```

```
*        ****************************************************************
*        **********                                            **********
*        **********   BIRTHDATE INDICATOR (F) TEST ROUTINE     **********
*        **********                                            **********
*        ****************************************************************
DOBCHK   CLI   0(7),C'F'             TEST CARD COL FOR BIRTHYEAR IND
         BC    7,YGCHK               COND BRANCH TO NEXT TEST IF NE
         LA    3,YGCHK               SAVE NEXT TEST ROUT ADDR IN R3
         LA    2,5                   ADD 5 TO CARD COL PTR
         CLI   1(7),C'='             TEST FOR EQUAL COMPARE INDICATOR
         BC    7,*+16                COND BRANCH AROUND EQUAL MASKING
         OI    FLGMASK,X'01'         SET ROUT INDICATOR TO = COMPARE
         LA    6,EQMASK              SAVE EQUAL MASK ADDR IN R6
         B     NUMTEST-20            BRANCH AROUND OTHER DOB TESTS
         CLI   1(7),C'#'             TEST FOR UNEQUAL COMPARE IND
         BC    7,NUMTEST-4           COND BRANCH TO RANGE TEST
         OI    FLGMASK,X'02'         SET ROUT IND TO UNEQUAL MASK
         LA    6,NEMASK              SAVE UNEQUAL MASK ADDR IN R6
         MVC   21(2,6),2(7)          MOVE BIRTHDATE YEAR TO MASK AREA
         MVC   88(2,6),ONES          MOVE 1'S INTO AND-MASK
         B     BLKCHK                BRANCH TO BLANK CARD COL TEST
         LA    6,RDTAPE+58           SAVE DOB ADDR IN TAPE READ AREA
*        ****************************************************************
*        **********                                            **********
*        ********** GREATER-THAN/LESS-THAN MASKING ROUTINE     **********
*        **********                                            **********
*        ****************************************************************
NUMTEST  OI    FLGMASK,X'04'         SET ROUT IND TO < AND > COMPARE
         LA    2,8                   ADD 8 TO COL CARD PTR
         CLI   NUMFLG,C'0'           TEST FOR PREVIOUS < AND > COMP
         BC    7,*                   COND BRANCH TO 2ND TEST
         MVI   NUMFLG,C'1'           SET FLG TO 1 TEST
         MVC   SAVEGT1(2),2(7)       MOVE DATA TO 1ST > TEST AREA
         MVC   SAVELT1(2),5(7)       MOVE DATA TO 1ST < TEST AREA
         LR    8,6                   SAVE DATA FIELD RECORD ADDR
         B     BLKCHK                BRANCH TO BLANK CARD COL TEST
         CLI   NUMFLG,C'$'           TEST FOR 2 < AND > COMPARES
         BC    8,ERROR               COND BRANCH TO ERROR ROUT IF MAX
         MVI   NUMFLG,C'$'           SET FLG TO MAX < AND > COMPARES
         MVC   SAVEGT2(2),2(7)       MOVE DATA TO 2ND > TEST AREA
         MVC   SAVELT2(2),5(7)       MOVE DATA TO 2ND < TEST AREA
         LR    9,6                   SAVE DATA FIELD RECORD ADDR
         B     BLKCHK                BRANCH TO BLANK CARD COL TEST
```

```
*        ************************************************************
*        **********                                        **********
*        **********   YEAR-GROUP INDICATOR (G) TEST ROUTINE **********
*        **********                                        **********
*        ************************************************************

YGCHK    CLI   0(7),C'G'              TEST CARD COL FOR YEAR GRP IND
         BC    7,RANKCHK              COND BRANCH TO NEXT TEST IF NE
         LA    3,RANKCHK              SAVE NEXT TEST ROUT ADDR IN R3
         LA    2,5                    ADD 5 TO CARD COL PTR
         CLI   1(7),C'='              TEST FOR EQUAL COMPARE INDICATOR
         BC    7,*+16                 COND BRANCH AROUND EQUAL MASKING
         OI    FLGMASK,X'01'          SET ROUT INDICATOR TO = COMPARE
         LA    6,EQMASK               SAVE EQUAL MASK ADDR IN R6
         B     RANKCHK-16             BRANCH AROUND OTHER YR GP TESTS
         CLI   1(7),C'#'              TEST FOR UNEQUAL COMPARE IND
         BC    7,*+16                 COND BRANCH AROUND # MASK ROUT
         OI    FLGMASK,X'02'          SET ROUT IND TO UNEQUAL MASK
         LA    6,NEMASK               SAVE UNEQUAL MASK ADDR IN R6
         B     RANKCHK-16             BRANCH AROUND YR GP RANGE TEST
         LA    6,RDTAPE+42            SAVE YRGP ADDR IN TAPE READ AREA
         B     NUMTEST                BRANCH TO < AND > MASK ROUTINE
         MVC   5(2,6),2(7)            MOVE YEAR GROUP INTO MASK AREA
         MVC   72(2,6),ONES           MOVE 1'S INTO AND MASK
         B     BLKCHK                 BRANCH TO BLANK CARD COL TEST

*        ************************************************************
*        **********                                        **********
*        **********      RANK INDICATOR (H) TEST ROUTINE   **********
*        **********                                        **********
*        ************************************************************

RANKCHK  CLI   0(7),C'H'              TEST CARD COL FOR RANK INDICATOR
         BC    7,CURCHK               COND BRANCH TO NEXT TEST IF NE
         LA    3,CURCHK               SAVE NEXT TEST ROUT ADDR IN R3
         OI    FLGMASK,X'01'          SET ROUT INDICATOR TO = COMPARE
         LA    2,4                    ADD 4 TO CARD COL PTR
         MVC   EQMASK(1),2(7)         MOVE RANK CODE INTO = MASK AREA
         MVI   EQMASK+67,X'FF'        MOVE 1'S INTO = AND MASK
         B     BLKCHK                 BRANCH TO BLANK CARD COL TEST

*        ************************************************************
*        **********                                        **********
*        **********  CURRICULUM INDICATOR (I) TEST ROUTINE **********
*        **********                                        **********
*        ************************************************************

CURCHK   CLI   0(7),C'I'              TEST CARD COL FOR CURRICULUM IND
```

```
                BC      7,DEGACHK               COND BRANCH TO NEXT TEST IF NE
                LA      3,DEGACHK               SAVE NEXT TEST ROUT ADDR IN R3
                OI      FLGMASK,X'01'           SET ROUT INDICATOR TO = COMPARE
                LA      2,6                     ADD 6 TO CARD COL PTR
                MVC     EQMASK+39(3),2(7)       MOVE CURRICULUM INTO = MASK AREA
                MVC     EQMASK+104(3),ONES      MOVE 1'S INTO = AND MASK
                B       BLKCHK                  BRANCH TO BLANK CARD COL TEST
*               ********************************************************************
*               **********                                              **********
*               **********   EDUCATIONAL LEVEL AND AREA INDICATORS      **********
*               **********          (J & K) TEST ROUTINES               **********
*               **********                                              **********
*               ********************************************************************
DEGACHK         CLI     0(7),C'J'               TEST CARD COL FOR EDUCATION DEG
                BC      7,DEGLCHK               COND BRANCH TO NEXT TEST IF NE
                LA      3,DEGLCHK               SAVE NEXT TEST ROUT ADDR IN R3
                OI      FLGMASK,X'01'           SET ROUT INDICATOR TO = COMPARE
                LA      2,4                     ADD 4 TO CARD COL PTR
                MVC     EQMASK+63(1),2(7)       MOVE DEG AREA CDE TO = MASK AREA
                MVI     EQMASK+130,X'FF'        MOVE 1'S INTO = AND MASK
                B       BLKCHK                  BRANCH TO BLANK CARD COL TEST
DEGLCHK         CLI     0(7),C'K'               TEST CARD COL FOR EDUCATION LEVL
                BC      7,QPRGCHK               COND BRANCH TO NEXT TEST IF NE
                LA      3,QPRGCHK               SAVE NEXT TEST ROUT ADDR
                OI      FLGMASK,X'01'           SET ROUT INDICATOR TO = COMPARE
                LA      2,4                     ADD 4 TO CARD COL PTR
                MVC     EQMASK+42(1),2(7)       MOVE DEG LEVEL CODE TO = MASK
                MVI     EQMASK+109,X'FF'        MOVE 1'S INTO = AND MASK
                B       BLKCHK                  BRANCH TO BLANK CARD COL TEST
*               ********************************************************************
*               **********                                              **********
*               **********     GRADUATE AND TOTAL QPR INDICATORS        **********
*               **********          (L & M) TEST ROUTINES               **********
*               **********                                              **********
*               ********************************************************************
QPRGCHK         CLI     0(7),C'L'               TEST CARD COL FOR GRAD QPR IND
                BC      7,QPRTCHK               COND BRANCH TO NEXT TEST IF NE
                LA      3,QPRTCHK               SAVE NEXT TEST ROUT ADDR
                LA      6,RDTAPE+94             SAVE QPRG ADDR IN TAPE READ AREA
                B       NUMTEST                 BRANCH TO < A' D > MASK ROUTINE
QPRTCHK         CLI     0(7),C'M'               TEST CARD COL FOR TOT QPR IND
                BC      7,GDTECHK               COND BRANCH TO NEXT TEST IF NE
                LA      3,GDTECHK               SAVE NEXT TEST ROUT ADDR
                LA      6,RDTAPE+97             SAVE QPRT ADDR IN TAPE READ AREA
```

```
         B       NUMTEST                     BRANCH TO < AND > MASK ROUTINE

*        ************************************************************
*        **********                                      **********
*        **********    GRADUATION YEAR INDICATOR (N) TEST **********
*        **********                ROUTINE               **********
*        **********                                      **********
*        ************************************************************

GDTECHK  CLI     0(7),C'N'                  TEST CARD COL FOR GRAD DATE IND
         BC      7,SSNRCHK                  COND BRANCH TO NEXT TEST IF NE
         LA      3,SSNRCHK                  SAVE NEXT TEST ROUT ADDR
         LA      6,RDTAPE+92                SAVE GDTE ADDR IN TAPE READ AREA
         CLI     1(7),C'='                  TEST FOR EQUAL COMPARE INDICATOR
         BC      7,NUMTEST                  BRANCH TO < AND > MASK ROUTINE
         OI      FLGMASK,X'01'              SET ROUT INDICATOR TO = COMPARE
         LA      2,5                        ADD 5 TO CARD COL PTR
         MVC     EQMASK+55(2),2(7)          MOVE GRAD DATE INTO = MASK AREA
         MVC     EQMASK+122(2),ONES         MOVE 1'S INTO = AND MASK
         B       BLKCHK                     BRANCH TO BLANK CARD COL TEST


*        ************************************************************
*        **********                                      **********
*        **********  INITIALIZE TAPE SEARCH AND DATA PRINT **********
*        **********            OUTPUT ROUTINE            **********
*        **********                                      **********
*        ************************************************************

MYEOF    LA      5,LINE                     SET R5 TO PRINT LINE OUTPUT ADDR
         LA      6,74                       SET PAGE LINE CT TO 70 LINES R6
         LA      7,RDTAPE                   SET R7 TO TAPE READ AREA ADDR
         PUT     OUT,TITL1                  PRINT 1ST LINE OF PAGE HEADING
         PUT     OUT,TITL2                  PRINT 2ND LINE OF PAGE HEADING
         PUT     OUT,TITL3                  PRINT 3RD LINE OF PAGE HEADING
         PUT     OUT,TITL4                  PRINT PAGE HEADING SEPARATER

*        ************************************************************
*        **********                                      **********
*        **********  DETERMINE TYPE OF TAPE SEARCH SPECIFIED **********
*        **********                                      **********
*        ************************************************************

         CLI     FLGMASK,X'07'              TEST COMPARE IND FOR = # < >
         BC      8,ROUT7                    COND BRANCH TO ALL COMPARE ROUT
         TM      FLGMASK,X'03'              TEST COMPARE IND FOR = AND #
```

```
         BC    1,ROUT3                  COND BRANCH TO EQUAL COMPARE
         BC    8,ROUT4                  COND BRANCH TO < AND COMPARE
         TM    FLGMASK,X'06'            TEST COMPARE IND FOR # > <
         BC    1,ROUT6                  COND BRANCH TO # < AND > COMPARE
         BC    8,ROUT1                  COND BRANCH TO EQUAL COMPARE
         CLI   FLGMASK,X'02'            TEST COMPARE IND FOR UNEQUAL
         BC    8,ROUT2                  COND BRANCH TO UNEQUAL COMPARE

*        ********************************************************************
*        ********************************************************************
*        **********                                          **********
*        **********          TAPE SEARCH ROUTINES            **********
*        **********                                          **********
*        ********************************************************************
*        ********************************************************************

ROUT5    GET   RTP,(7)                  READ TAPE RECORD
         LR    7,1                      RESET R7 TO TAPE READ AREA ADDR
         LA    4,ROUT5                  SAVE RETURN ADDR IN R4
         MVC   SAVEQ(67),37(7)          MOVE TAPE REC DATA TO =SAVE AREA
         NC    SAVEQ(67),EQMASK+67      REMOVE NON TESTED TAPE REC DATA
         CLC   SAVEQ(67),EQMASK         TEST TAPE DATA FOR EQUAL COMPARE
         BC    7,0(4)                   COND BRANCH TO READ TAPE INST

*        ********************************************************************
*        **********                                          **********
*        **********  GREATER-THAN/LESS-THAN SEARCH ROUTINE   **********
*        **********                                          **********
*        ********************************************************************

NUMCHK   CLI   NUMFLG,C'$'              TEST RANGE COMPARE FOR 2 COMPS
         BC    7,*+24                   COND BRANCH AROUND 2ND COMPARE
         CLC   0(2,9),SAVEGT2           TEST TAPE DATA FOR > COMPARE (1)
         BC    4,0(4)                   COND BRANCH TO READ TAPE INST
         CLC   0(2,9),SAVELT2           TEST TAPE DATA FOR < COMPARE (1)
         BC    2,0(4)                   COND BRANCH TO READ TAPE INST
         CLC   0(2,8),SAVEGT1           TEST TAPE DATA FOR > COMPARE (2)
         BC    4,0(4)                   COND BRANCH TO READ TAPE INST
         CLC   0(2,8),SAVELT1           TEST TAPE DATA FOR 2 COMPARE (2)
         BC    2,0(4)                   COND BRANCH TO READ TAPE INST
         B     PRTDATA                  BRANCH TO PRINT RECORD ROUTINE
ROUT6    GET   RTP,(7)                  READ TAPE RECORD
         LR    7,1                      RESET R7 TO TAPE READ AREA ADDR
         LA    4,ROUT6                  SAVE RETURN ADDR IN R4
         MVC   SAVNE(67),37(7)          MOVE TAPE REC DATA TO #SAVE AREA
         NC    SAVNE(67),NEMASK+67      REMOVE NON-TESTED TAPE REC DATA
         CLC   SAVNE(67),NEMASK         TEST TAPE FOR UNEQUAL COMPARE
```

```
         BC     8,0(4)                  COND BRANCH TO READ TAPE INST
         B      NUMCHK                  BRANCH TO < AND > COMPARE ROUT
ROUT7    GET    RTP,(7)                 READ TAPE DATA
         LR     7,1                     RESET R7 TO TAPE READ AREA ADDR
         LA     4,ROUT7                 SAVE RETURN ADDR IN R4
         MVC    SAVEQ(67),37(7)         MOVE TAPE REC DATA TO =SAVE AREA
         NC     SAVEQ(67),EQMASK+67     REMOVE NON TESTED TAPE REC DATA
         CLC    SAVEQ(67),EQMASK        TEST TAPE DA   FOR EQUAL COMPARE
         BC     7,0(4)                  COND BRANCH TO READ TAPE INST
         MVC    SAVNE(67),37(7)         MOVE TAPE REC DATA TO #SAVE AREA
         NC     SAVNE(67),NEMASK+67     REMOVE NON-TESTED TAPE REC DATA
         CLC    SAVNE(67),NEMASK        TEST TAPE FOR UNEQUAL COMPARE
         BC     8,0(4)                  COND BRANCH TO READ TAPE INST
         B      NUMCHK                  BRANCH TO < AND > COMPARE ROUT
ROUT1    GET    RTP,(7)                 READ TAPE RECORD
         LR     7,1                     RESET R7 TO TAPE READ AREA ADDR
         LA     4,ROUT1                 SAVE RETURN ADDR IN R4
         MVC    SAVEQ(67),37(7)         MOVE TAPE REC DATA TO =SAVE AREA
         NC     SAVEQ(67),EQMASK+67     REMOVE NON TESTED TAPE REC DATA
         CLC    SAVEQ(67),EQMASK        TEST TAPE DATA FOR EQUAL COMPARE
         BC     7,0(4)                  COND BRANCH TO READ TAPE INST
         B      PRTDATA                 BRANCH TO PRINT RECORD ROUTINE
ROUT2    GET    RTP,(7)                 READ TAPE RECORD
         LR     7,1                     RESET R7 TO TAPE READ AREA ADDR
         LA     4,ROUT2                 SAVE RETURN ADDR IN R4
         MVC    SAVNE(67),37(7)         MOVE TAPE REC DATA TO #SAVE AREA
         NC     SAVNE(67),NEMASK+67     REMOVE NON-TESTED TAPE REC DATA
         CLC    SAVNE(67),NEMASK        TEST TAPE FOR UNEQUAL COMPARE
         BC     8,0(4)                  COND BRANCH TO READ TAPE INST
         B      PRTDATA                 BRANCH TO PRINT RECORD ROUTINE
ROUT3    GET    RTP,(7)                 READ TAPE RECORD
         LR     7,1                     RESET R7 TO TAPE READ AREA ADDR
         LA     4,ROUT3                 SAVE RETURN ADDR IN R4
         MVC    SAVEQ(67),37(7)         MOVE TAPE REC DATA TO =SAVE AREA
         NC     SAVEQ(67),EQMASK+67     REMOVE NON TESTED TAPE REC DATA
         CLC    SAVEQ(67),EQMASK        TEST TAPE DATA FOR EQUAL COMPARE
         BC     7,0(4)                  COND BRANCH TO READ TAPE INST
         MVC    SAVNE(67),37(7)         MOVE TAPE REC DATA TO #SAVE AREA
         NC     SAVNE(67),NEMASK+67     REMOVE NON-TESTED TAPE REC DATA
         CLC    SAVNE(67),NEMASK        TEST TAPE FOR UNEQUAL COMPARE
         BC     8,0(4)                  COND BRANCH TO READ TAPE INST
         B      PRTDATA                 BRANCH TO PRINT RECORD ROUTINE
ROUT4    GET    RTP,(7)                 READ TAPE RECORD
         LR     7,1                     RESET R7 TO TAPE READ AREA ADDR
         LA     4,ROUT4                 SAVE RETURN ADDR IN R4
         B      NUMCHK                  BRANCH TO < AND > COMPARE ROUT
```

```
*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********  PRINT TITLE AND SET DATA FLAG SWITCHES        **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************

TITLE    PUT   OUT,TITL1                 PRINT 1ST LINE OF PAGE HEADING
         PUT   OUT,TITL2                 PRINT 2ND LINE OF PAGE HEADING
         PUT   OUT,TITL3                 PRINT 3RD LINE OF PAGE HEADING
         PUT   OUT,TITL4                 PRINT PAGE HEADING SEPARATER
         LA    6,74                      SET LINE CTR (R6) TO 70 LINES
         B     EDIT                      BRANCH AROUND DATA TEST AREA
PRTDATA  MVC   TEST,103(7)               MOVE DUTY STA NR INTO TEST AREA
         MVI   DECK,C' '                 CLEAR 2 CARD DATA DECK FLAG
         NI    TEST,X'0F'                CONVERT DUTY STA CODE TO BINARY
         SR    12,12                     CLEAR R12
         IC    12,TEST                   INSERT BINARY NR IN R12
         CLI   103(7),C'0'               TEST MIN VALUE OF DUTY STA NR
         BC    8,*+26                    COND BRANCH IF FOREIGN STUDENT
         CLI   103(7),C'3'               TEST FOR 2 CARD DECK
         BC    2,*+8                     COND BRANCH IF NOT 2 CARD DECK
         MVI   DECK,C'*'                 SET 2 CARD DECK FLAG
         CP    12,6                      TEST LINES REMAINING ON PAGE
         BC    2,TITLE                   COND BRANCH TO NEW PAGE
         B     EDIT                      BRANCH AROUND FOREIGN FLAG ENTRY
         LA    2,2                       CLEAR R2
         CP    2,6                       TEST FOR 2 LINES LEFT ON PAGE
         BC    2,TITLE                   COND BRANCH TO NEW PAGE

*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********  'LOAD EDIT AND PRINT' ROUTINE FOR LINE        **********
*        **********            ONE DATA OUTPUT                     **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************

EDIT     MVI   0(5),C' '                 INSERT BLANK CHAR IN OUTPUT LINE
         MVC   1(132,5),0(5)             CLEAR OUTPUT LINE
         MVI   0(5),C'0'                 INSERT DOUBLE SPACE CHARACTER
         MVI   1(31,5),6(7)              MOVE NAME INTO OUTPUT LINE
         MVC   61(6,5),48(7)             MOVE DOR INTO OUTPUT LINE
         MVC   45(6,5),58(7)             MOVE DOB INTO OUTPUT LINE
         CLI   101(7),C'1'               TEST FOR NON-U.S.OFFICER
```

```
         BC     2,FPST                    COND BRANCH AROUND U.S. ENTRIES
         MVC    33(9,5),65(7)             MOVE SSNR INTO OUTPUT LINE
         CLI    10(7),C'0'                TEST FOR USN OFFICER
         BC     7,FPST                    COND BRANCH AROUND NON-USN ENTRY
         MVC    54(4,5),38(7)             MOVE DESIGNATOR INTO OUTPUT LINE
         MVC    70(2,5),45(7)             MOVE SOURCE CODE INTO OUTPT LINE
         MVI    72(5),C'-'                INSERT SPACER
         MVC    73(1,5),47(7)             MOVE SRCE CODE MOD TO OUTPT LINE
         MVC    77(2,5),42(7)             MOVE YEAR GROUP INTO OUTPUT LINE
         MVC    82(2,5),74(7)             MOVE 1ST RET YR INTO OUTPUT LINE
         MVC    87(4,5),54(7)             MOVE P-CODE INTO OUTPUT LINE
FPST     CLI    88(7),C' '                TEST FOR START YEAR ENTRY
         BC     8,*+20                    COND BRANCH AROUND START YEAR
         MVC    94(1,5),88(7)             MOVE START QTR INTO OUTPUT LINE
         MVI    95(5),C'-'                INSERT SPACER
         MVC    96(2,5),89(7)             MOVE START YR INTO OUTPUT LINE
         MVC    101(3,5),76(7)            MOVE CURRICULUM INTO OUTPUT LINE

*        ********************************************************************
*        **********                                                **********
*        **********          EDUCATIONAL LEVEL CHECK               **********
*        **********                                                **********
*        ********************************************************************

         NI     79(7),X'0F'               CONVERT DEGR CODE TO BINARY NR
         SR     2,2                       CLEAR R2
         IC     2,79(7)                   INSERT DEGR CODE IN R2
         SLL    2,2                       MULTIPLY DEGR CODE BY 4
         LA     3,TABLE1                  PUT DEGR TABLE ADDR IN R4
         AP     2,3                       SET P2 TO CORRECT DEGR ADDR
         MVC    106(4,5),0(2)             MOVE DEGR INTO OUTPUT LINE
         MVC    112(3,5),94(7)            MOVE GRAD QTR INTO OUTPUT LINE
         MVC    117(16,5),104(7)          MOVE 1ST DUSTA INTO OUTPUT LINE
         SR     12,10                     SUBTRACT 1 FROM NR OF DUTY STAS
         LA     2,2                       INSERT CONSTANT 2 IN REG 2
         SR     6,2                       SUBTRACT 2 FROM LINE CTR
         PUT    OUT,LINE                  PRINT 1ST LINE OF OUTPUT

*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********  'LOAD EDIT AND PRINT'ROUTINE FOR LINE         **********
*        **********          TWO DATA OUTPUT                       **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************
```

```
         MVI   0(5),C' '                 INSERT BLANK CHAR IN OUTPUT LINE
         MVC   1(132,5),0(5)             CLEAR OUTPUT LINE
*        ********************************************************************
*        **********                                              **********
*        **********            RANK CODE CHECK                   **********
*        **********                                              **********
*        ********************************************************************
         SR    2,2                       CLEAR R2
         MVC   TEST,37(7)                MOVE RANK CODE INTO TEST AREA
         NI    37(7),X'0F'               CONVERT RANK CODE TO BINARY NR
         IC    2,37(7)                   INSERT BINARY NR IN R2
         SLL   2,2                       MULTIPLY NR BY 4
         TM    TEST,X'30'                TEST RANK CODE FOR A-I
         BC    8,*+28                    COND BRANCH TO A-I RK TABLE SEG
         TM    TEST,X'10'                TEST RANK CODE FOR J-R SEGMENT
         BC    8,*+12                    COND BRANCH TO S-Z RK TABLE SEG
         LA    3,TABLEG+32               SET R3 TO J-R RANK TABLE SEGMENT
         B     *+16                      BRANCH TO RK TABLE POINTER ROUT
         LA    3,TABLEG+64               SET R3 TO S-Z RANK TABLE SEGMENT
         B     *+8                       BRANCH TO RK TABLE POINTER ROUT
         LA    3,TABLEG-4                SET R3 TO A-I RANK TABLE SEGMENT
         AR    3,2                       SET R3 TO CORRECT RANK ADDRESS
         MVC   1(4,5),0(3)               MOVE RANK INTO OUTPUT LINE
*        ********************************************************************
*        **********                                              **********
*        ********** SERVICE NUMBER AND FOREIGN COUNTRY CHECK     **********
*        **********                                              **********
*        ********************************************************************
         SR    2,2                       CLEAR R2
         CLI   101(7),X'F2'              TEST FOR RANK CODE
         BC    4,SVC                     COND BRANCH TO SVC DECODING ROUT
         NI    101(7),X'0F'              CONVERT 1ST CTY CDE DIGIT TO BIN
         NI    102(7),X'0F'              CONVERT 2ND CTY CDE DIGIT TO BIN
         LA    3,TABLEC-160              SET R4 AS CTY TABLE POINTER
         IC    2,101(7)                  INSERT 1ST CTY CODE DIGIT IN R2
         LA    10,80                     SET R10 TO 80
         AR    3,10                      ADD 10 LINES TO CTY POINTER
         BCT   2,*-2                     INCREASE POINTER VALUE IF REQRD
         IC    2,102(7)                  INSERT 2ND CTY CODE DIGIT IN R2
         SLL   2,3                       MULTIPLY R2 BY 8
         AR    3,2                       SET CTY TABLE ADDR POINTER (R4)
         MVC   13(7,5),0(3)              MOVE CTY NAME INTO OUTPUT LINE
         LA    10,1                      SET CONSTANT 1 IN R10
         B     SFX                       BRANCH TO NEXT OUTPUT LINE ENTRY
```

```
SVC      CLC   101(2,7),=C'01'          TEST FOR " IGEP " ENTRY
         BC    7,*+8                    COND BRANCH AROUND IGEP ENTRY
         MVI   30(5),X'C9'              MOVE I INTO OUTPUT LINE
         NI    102(7),X'0F'             CONVERT SVC CODE TO BINARY NR
         IC    2,102(7)                 INSERT NR IN R2
         SLL   2,2                      MULTIPLY NR BY 4
         CLI   101(7),C'0'              TEST FOR NON-US OFFICER
         BC    8,*+12                   COND BRANCH TO USN TABLE
         LA    3,TABLES+12              SET R3 TO NON-USN POINTER
         B     *+22                     BRANCH AROUND USN ENTRIES
         MVI   5(5),C','                INSERT COMMA AFTER RANK
         MVC   6(6,5),0(7)              MOVE SERIAL NR INTO OUTPUT LINE
         MVI   12(5),C','               INSERT COMMA
         LA    3,TABLES                 SET R3 TO SVC TABLE BASE ADDR
         AR    3,2                      SET SVC TABLE ADDR POINTER (R3)
         MVC   13(4,5),0(3)             MOVE SVC INTO OUTPUT LINE
SEX      CLI   64(7),X'C6'              TEST FOR FEMALE SEX INDICATOR
         BC    7,*+8                    COND BRANCH TO DESIG CHG ENTRY
         MVI   30(5),X'C6'              MOVE F INTO OUTPUT LINE
         CLI   80(7),C' '               TEST FOR DESIGNATOR CHANGE ENTRY
         BC    8,*+18                   COND BRCH AROUND OLD DESIG DATA
         MVI   53(5),C'('               INSERT OPENING PAREN
         MVC   54(4,5),80(7)            MOVE DESIG CHG DTE IN OUTPT LINE
         MVI   58(5),C')'               INSERT CLOSING PAREN
         MVC   94(1,5),91(7)            MOVE END QTR INTO OUTPUT LINE
         MVI   95(5),C'-'               INSERT SPACER
         MVC   96(2,5),92(7)            MOVE END YEAR INTO OUTPUT LINE

*        ********************************************************************
*        **********                                              **********
*        **********              DEGREE AREA CHECK               **********
*        **********                                              **********
*        ********************************************************************

         SR    2,2                      CLEAR R2
         MVC   TEST,100(7)              MOVE AREA CODE INTO TEST AREA
         NI    100(7),X'0F'             CONVERT AREA CODE TO BINARY NR
         IC    2,100(7)                 INSERT BINARY NR IN R2
         SLL   2,2                      MULTIPLY NR BY 4
         TM    TEST,X'30'               TEST DEG AREA CDE FOR A-I SEG
         BC    8,*+28                   COND BRANCH TO A-I DEG TABLE SEG
         TM    TEST,X'10'               TEST DEG AREA CDE FOR J-R SEG
         BC    8,*+12                   COND BRANCH TO S-Z DEG TABLE SEG
         LA    3,TABLED+32              SET R3 TO DEG TABLE SEGMENT
         B     *+16                     BRANCH TO TABLE PTR ROUTINE
         LA    3,TABLED+64              SET R3 TO S-Z DEG TABLE SEGMENT
         B     *+8                      BRANCH TO TABLE PTR ROUTINE
         LA    3,TABLED-4               SET R3 TO A-I DEG TABLE SEGMENT
```

```
         AR    3,2                     SET AREA TABLE PTR TO CODE ADDR
         MVC   106(4,5),0(3)           MOVE DEGR AREA INTO OUTPUT LINE
         MVC   112(3,5),97(7)          MOVE TOTAL OPR INTO OUTPUT LINE
         MVC   117(16,5),120(7)        MOVE 2ND DUTYSTA INTO OUTPT LINE
         SR    12,10                   SUBTRACT 1 FROM NR OF DUTY STAS
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,LINE                PRINT 2ND LINE OF OUTPUT

*        ********************************************************************
*        ********************************************************************
*        **********                                              **********
*        **********  'LOAD EDIT AND PRINT'ROUTINE FOR LINE       **********
*        **********              THREE DATA OUTPUT               **********
*        **********                                              **********
*        ********************************************************************
*        ********************************************************************

         CLI   103(7),C'0'             TEST FOREIGN STUDENT FLAG
         BC    8,0(4)                  COND BRANCH TO READ NEW  REC
         MVI   0(5),C' '               INSERT BLANK CHAR IN OUTPUT LINE
         MVC   1(132,5),0(5)           CLEAR OUTPUT LINE
         CLI   84(7),C' '              TEST FOR DESIGNATOR CHG DATA
         BC    7,*+14                  COND BRANCH TO OLD DESIG ENTRY
         CR    12,10                   TEST NR OF DUTY STAS LEFT TO PRT
         BC    4,0(4)                  COND BRANCH TO READ NEW RECORD
         B     *+18                    BRANCH TO 3RD DUTY STA INPUT
         MVI   53(5),C'('              INSERT OPENING PAREN
         MVC   54(4,5),84(7)           MOVE OLD DESIG INTO OUTPUT LINE
         MVI   58(5),C')'              INSERT CLOSING PAREN
         MVC   117(16,5),136(7)        MOVE 3RD DUTYSTA INTO OUTPT LINE
         SR    6,10                    SUBTRACT 1 FROM LINE CTR
         PUT   OUT,LINE                PRINT 3RD LINE OF OUTPUT

*        ********************************************************************
*        ********************************************************************
*        **********                                              **********
*        **********  'LOAD EDIT AND PRINT'ROUTINE FOR LINE       **********
*        **********     FOUR THROUGH EIGHT DATA OUTPUT           **********
*        **********                                              **********
*        ********************************************************************
*        ********************************************************************

         CLI   DECK,C'*'               TEST FOR 2 CARD DATA DECK FLAG
         BC    8,0(4)                  COND BRANCH TO TAPE READ ROUTINE
         SR    12,10                   SUBTRACT 1 FROM NR OF DUTY STAS
         BC    13,0(4)                 COND BRANCH TO TAPE READ ROUTINE
```

```
         LA    2,152(7)                  MOVE NEXT DUTY STA ADDR INTO R2
         MVI   0(5),C' '                 INSERT BLANK CHAR IN OUTPUT LINE
         MVC   1(132,5),0(5)             CLEAR OUTPUT LINE
         MVC   117(16,5),0(2)            MOVE NEXT DUTYSTA TO OUTPUT LINE
         SR    6,10                      SUBTRACT 1 FROM LINE CTR
         PUT   OUT,LINE                  PRINT NEXT OUTPUT LINE
         LA    2,16(2)                   SET DUTY STA POINTER TO NEXT STA
         BCT   12,*-36                   COND BRANCH TO DUTSTA PRT ROUT
         B     0(4)                      BRANCH TO TAPE READ ROUTINE

*        ********************************************************************
*        **********                                                **********
*        **********                                                **********
*        **********          PROGRAM CLOSE ROUTINE                 **********
*        **********                                                **********
*        ********************************************************************

ERROR    PUT   OUT,ERRLINE               PRINT CARD DATA ERROR MESSAGE
MYEOF1   PUT   OUT,END                   PRINT END OF DATA MESSAGE
         CLOSE (IN,,RTP,,OUT)
         L     13,4(13)
         RETURN (14,12),T

*        ********************************************************************
*        ********************************************************************
*        **********                                                **********
*        **********    TABLES,DEFINED CONSTANTS,RESERVED           **********
*        **********  STORAGE AREAS AND INPUT/OUTPUT DATA           **********
*        **********              DATA CONTROL                      **********
*        **********                                                **********
*        ********************************************************************
*        ********************************************************************

*        ********************************************************************
*        **********                                                **********
*        **********                                                **********
*        **********          FOREIGN COUNTRY TABLE                 **********
*        **********                                                **********
*        ********************************************************************

TABLFC   DC    C'BRAZIL  '
         DC    C'CANADA  '
         DC    C'CEYLON  '
         DC    C'CHILE   '
         DC    C'CHINA   '
         DC    C'COLOMBIA'
         DC    C'ECUADOR '
         DC    C'GERMANY '
         DC    C'INDIA   '
```

```
         DC    C'IRAN    '
         DC    C'ISRAEL  '
         DC    C'KOREA   '
         DC    C'NORWAY  '
         DC    C'PERU    '
         DC    C'PHILL.  '
         DC    C'PORT.   '
         DC    C'THAILAND'
         DC    C'TURKEY  '
         DC    C'URUAGUAY'
         DC    C'VENE.   '
         DC    C'VIETNAM '
         DC    C'FORIGN  '
*        ****************************************************
*        **********                                **********
*        **********   DEGREE (EDUCATIONAL) AREA TABLE **********
*        **********                                **********
*        ****************************************************
TABLED   DC    C'NONE'
         DC    C'A.E.'
         DC    C'AENG'
         DC    C'ESCI'
         DC    C'I.R.'
         DC    C'E.E.'
         DC    C'EENG'
         DC    C'METO'
         DC    C'OCEN'
         DC    C'C.R.'
         DC    C'CMGT'
         DC    C'C.S.'
         DC    C'MGT.'
         DC    C'M.E.'
         DC    C'MENG'
         DC    C'MATH'
         DC    C'PHY '
         DC    C'CHEM'
         DC    C'MATS'
         DC    C'C.E.'
*        ****************************************************
*        **********                                **********
*        **********      RANK AND TITLE TABLE      **********
*        **********                                **********
*        ****************************************************
TABLEG   DC    C'EADM'
```

```
         DC     C'ADM '
         DC     C'VADM'
         DC     C'RADM'
         DC     C'RADM'
         DC     C'COMO'
         DC     C'CAPT'
         DC     C'CDR '
         DC     C'LCDR'
         DC     C'LT  '
         DC     C'LTJG'
         DC     C'ENS '
         DC     C'CIV '
         DC     C'2LT '
         DC     C'1LT '
         DC     C'MAJ '
         DC     C'LCOL'
         DC     C'COL '
         DC     C'GEN '
*        ************************************************************
*        **********                                      **********
*        **********          EDUCATIONAL LEVEL TABLE     **********
*        **********                                      **********
*        ************************************************************
TABLEL   DC     C'GSCH'
         DC     C'HSCH'
         DC     C'1YRC'
         DC     C'2YRC'
         DC     C'3YRC'
         DC     C' PG '
         DC     C'BACH'
         DC     C'CERT'
         DC     C' MS '
         DC     C'PHD '
*        ************************************************************
*        **********                                      **********
*        **********          ARMED FORCES TABLE          **********
*        **********                                      **********
*        ************************************************************
TABLFS   DC     C'USN '
         DC     C'USN '
         DC     C'USNR'
         DC     C'USA '
         DC     C'USCG'
         DC     C'USMC'
```

```
         DC    C'USAF'
*        ******************************************************************
*        **********                                              **********
*        ********** DATA STATUS MESSAGES AND TITLE CONSTANTS     **********
*        **********                                              **********
*        ******************************************************************
TITL1    DC    C'1NAME
                     RANK     ORIG          1ST        SOCIAL        BIRTH    *
                                                           NPGS      CUR  DEGR*
TITL2    DC    C' RANK,FILE NR,SVC/CTY(,*)
               DFSIG    DATE   SOURCE   YR    RET    P SECURITY    DATE      *
                 OPR                                     QTR-YR    PIC     / *
TITL3    DC    C' (*=F IF FEMALE,=I IF IGEP)
               (CHG)  YR-M-D   CODE    GP    YR    CODE   NUMBER   YR-M-D    *
                 G/T  PAST DUTY STAS                        ST/END  LUM  AREA*
TITL4    DC    C'+___________________________________     __________  ______*
               ____  ________  _______ ___T__   ___  ____  _______  ___  ____*
ERRLINE  DC    C'0********     ERROR IN INPUT CARDS     **********    PRO*
               GRAM WILL NOT RUN    **********     RECHECK INPUT CARDS A*
               ND RESUBMIT      *********'
END      DC    C'0##########                END OF DATA         ########## *
                          END OF DATA         ##########            END OF *
               DATA             #########'
*        ******************************************************************
*        **********                                              **********
*        **********            DEFINED STORAGE AREA              **********
*        **********                                              **********
*        ******************************************************************
DBCK     DC    X'00'                       TWO CARD DECK FL
TEST     DC    X'00'                       SCRATCH TEST AREA
ONES     DC    10X'FF'                     BINARY ONES STORAGE
FLGMASK  DC    X'00'                       COMPARE ROUTINE INDICATOR
NUMFLG   DC    C'0'                        DATA RANGE COMPARE INDICATOR
EQMASK   DC    134X'00'                    EQUAL COMPARE MASK
NEMASK   DC    134X'00'                    NOT EQUAL COMPARE MASK
CDAREA   DC    80X'00'                     CARD INPUT AREA
RDTAPE   DC    232X'00'                    TAPE INPUT AREA
LINE     DC    133X'00'                    OUTPUT DATA AREA
SAVEGT1  DS    CL2                         1ST GREATER THAN SAVE AREA
SAVELT1  DS    CL2                         1ST LESS THAN SAVE AREA
SAVEGT2  DS    CL2                         2ND GREATER THAN SAVE AREA
```

```
SAVELT2  DS    CL2                        2ND LESS THAN SAVE AREA
SAVEQ    DS    CL67                       DATA SAVE AREA FOR EQUAL TEST
SAVNE    DS    CL67                       DATA SAVE AREA FOR UNEEVAL TEST

*        ********************************************************************
*        **********                                                **********
*        **********     INPUT/OUTPUT DATA CONTROL BLOCKS           **********
*        **********                                                **********
*        ********************************************************************

IN       DCB   DDNAME=MASSAIN,DSORG=PS,MACRF=(GT),BLKSIZE=80,RECFM=F,E*
               ODAD=MYEOF,BUFNO=3

OUT      DCB   DDNAME=MASSAOUT,DSORG=PS,MACRF=(PM),BLKSIZE=133,RECFM=FA

RTP      DCB   DDNAME=MASRTAPE,DSORG=PS,MACRF=(GT),LRECL=232,BLKSIZE=92*
               8,RECFM=FB,EODAD=MYEOF1

         LTORG
         END

//GO.MASSAOUT DD SYSOUT=A
//GO.MASRTAPE DD DSNAME=HEN.DONE,UNIT=2400,VOLUME=SER=NPS194,DISP=(OLD,
//         KEEP),LABEL=(1,SL)
//GO.MASSAIN DD *
```

APPENDIX C

STATUS OF FILES AND PROGRAMS

The following information is provided to assist file and program maintenance personnel in carrying out assigned responsibilities.

SOIRS files are resident on labeled magnetic tape in fixed block record format with a blocking factor of four (4). Pertinent data on each file is listed as follows:

FILE	DSNAME	VOLUMN NUMBER
Bureau Original	HEN.SORT	NPS 209
Historical Father	HEN.MAS	NPS 138
Current Father	HEN.DONE	NPS 194
Historical Son	HEN.HSON	NPS 150
Current Son	HEN.CSON	NPS 155

SOIRS application programs are resident on disk pack (2311) FAC001 in load module form. The DSNAME of the library is HEN.-MISLIB. The member names of programs in this library correspond to the program names used in the main text of this paper and are listed as follows:

1. BUPSXFR
2. CRDCHK1
3. UPDATE
4. FILESRCH

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Naval Postgraduate School Monterey, California 93940	Unclassified
	2b. GROUP

3 REPORT TITLE

A Proposed Student Officer Information Retrieval System for the Naval Postgraduate School

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)

Master's Thesis: June 1969

5 AUTHOR(S) (First name, middle initial, last name)

Robert Lee Henry

Lawrence Lee Massa

6 REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO OF REFS
June 1969	147	0

8a. CONTRACT OR GRANT NO	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO	
c.	9b OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10 DISTRIBUTION STATEMENT

Distribution of this document is unlimited

11 SUPPLEMENTARY NOTES	12 SPONSORING MILITARY ACTIVITY
	Naval Postgraduate School Monterey, California 93940

13 ABSTRACT

Pertinent information on past and present Naval Postgraduate School students is now maintained, stored and processed in bulk files by Curriculum Officers. Information desired for management studies or analysis requires manual sorting of an ever increasing number of individual student records. This is an inadequate and inefficient system.

The foregoing problem could be resolved by the implementation of the Student Officer Information Retrieval System (SOIRS), which is a narrow scope retrieval system specifically designed to be responsive to the Curriculum Officer's needs with respect to student information. SOIRS evolved through a series of logical system design steps, identified as follows: (1) Problem Analysis; (2) Design of Records, Files and Reports; (3) Software Design; (4) Test of Entire System.

SOIRS is a complete system, establishing required files, updating files, and retrieving stored information

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N [illegible]

Security Classification

A-31408

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Information Retrieval						
Information System						
Management Information System						